DOCTOR OF SCIENCES

Multistream scheduling in cellular networks as a coupled optimization problem

Vanderpypen, Joël

Award date:
2012

Awarding institution:
University of Namur

Facultés Universitaires Notre-Dame de la Paix
Faculté d'Informatique
Namur, Belgique

Multistream Scheduling in Cellular Networks as a Coupled Optimization Problem

Joël Vanderpypen

January 2012

Thèse présentée en vue de l'obtention du grade de Docteur en Sciences
(orientation Informatique)

Ph.D. Supervisor:
Professor Laurent Schumacher,
University of Namur, Belgium.

Ph.D. Committee:
Professor Troels B. Sørensen,
Aalborg Universitet, Denmark.

Professor Luc Vandendorpe,
UCL, Belgium.

Opponents:
Professor Naji Habra,
University of Namur, Belgium.

Professor Jean-Paul Leclerq,
University of Namur, Belgium.

The public Ph.D. defense was held on the 24th January 2012 at the University of Namur.

Rempart de la Vierge, 13
B - 5000 Namur (Belgique)

Imprimé en Belgique
ISBN : 978-2-87037 - 750 - 5
Dépôt légal: D / 2012 / 1881 /9

Abstract

Cellular networks are more and more wide spread nowadays, moving from the current third generation UMTS to the forthcoming WiMAX and LTE 4G networks, promising ever increasing throughputs for new multimedia services. This thesis focuses on scheduling in these cellular networks, and proposes some multistream enhancements. Indeed, an efficient allocation of the available radio resources is one of the key parameters to provide larger throughputs to an ever increasing amount of users. To improve efficiency, multistream transmissions, where multiple users are allowed to transmit simultaneously, have been suggested. The resource allocation problem therefore becomes coupled through interferences, which we have to minimize. Moreover, this problem has multiple contradictory objectives, such as throughput, fairness, or respect of Quality of Service (QoS) guarantees. Both 3G's CDMA and 4G's OFDMA multiple access techniques are studied, for both SISO and MIMO transmissions. We have managed to decouple the resource allocation problem and limit interferences in CDMA through orthogonality factors, depending on users' channel dispersion in SISO, and on users' location in beamforming MIMO. As far as OFDMA is concerned, we have designed a priority-based scheduling algorithm which ensures some QoS guarantees, while remaining quite simple and requiring only a limited feedback, averaged over the full bandwidth. A MIMO extension of the algorithm, improving frequency reuse through a GoB strategy, has also been developed. The problem of WiMAX burst mapping has been deeply investigated, and a new mapping algorithm, seriously reducing resource wastes, has been designed.

Avant toute chose, je tiens à exprimer ma gratitude envers mon promoteur, Laurent Schumacher, qui m'a accueilli en faculté d'Informatique, et m'a offert l'opportunité de cette recherche doctorale. Son encadrement, ses conseils, ainsi que ses nombreux commentaires m'ont permis de mener à terme ce travail de longue haleine. Merci Laurent.

Je tiens également à remercier Troels Bundgaard Sørensen et Luc Vandendorpe, membres de mon comité d'accompagnement, ainsi que les autres membres de mon jury, Naji Habra et Jean-Paul Leclercq, pour le temps qu'ils ont consacré à me relire, ainsi que pour leurs remarques constructives et leurs conseils pertinents, me permettant d'améliorer sensiblement la qualité de ce travail.

Merci aussi à mes autres collègues, de bureau, de minifoot ou de pause-café, pour l'atmosphère de travail dont j'ai pu bénéficier, aux secrétaires pour leur disponibilité, ainsi qu'au F.R.I.A. et aux F.U.N.D.P. pour leur soutien financier.

Par ailleurs , je tiens également à adresser un merci tout spécial à Christie Freeman, pour le temps qu'elle a passé à corriger l'anglais perfectible de mon manuscrit, et sans laquelle la qualité linguistique de ce document ne serait pas ce qu'elle est maintenant.

Je veux enfin penser à ma famille et mes proches, au groupe de jeunes UJEB ainsi qu'au Club Nouvelle MINI, et les remercier pour leur présence, leurs encouragements et leur soutien tout au long de ses années. À ceux-là, et à tous ceux que j'oublie, un grand merci !

Joël, Décembre 2011

Contents

List of Figures

List of Tables

List of Algorithms

List of Acronyms

3GPP	Third Generation Partnership Project
AP	Access Point
BE	Best Effort
BER	Bit Error Rate
BS	Base Station
CBR	Constant Bit Rate
CDI	Channel Direction Indication
CDMA	Code Division Multiple Access
CQI	Channel Quality Indicator
CSI	Channel State Information
DL	Downlink
DSL	Digital Subscriber Line
EDF	Earliest Deadline First
EDGE	Enhanced Data for GSM Evolution
FCH	Frame Control Header
FDD	Frequency Division Duplex
FUSC	Full Usage of Subchannels
GBR	Guaranteed Bit Rate
GoB	Grid of Beams

GPRS	General Packet Radio Service
GSM	Global System for Mobile Communications
HSDPA	High Speed Downlink Packet Access
HSPA	High Speed Packet Access
HSUPA	High Speed Uplink Packet Access
IEEE	Institute of Electrical and Electronics Engineers
ISI	Intersymbol Interference
LAN	Local Area Network
LTE	Long Term Evolution
MAI	Multiple Access Interference
MCS	Modulation and Coding Scheme
MIMO	Multiple Input Multiple Output
MISO	Multiple Input Single Output
OFDM	Orthogonal Frequency Division Multiplexing
OFDMA	Orthogonal Frequency Division Multiple Access
PAPR	Peak to Average Power Ratio
PF	Proportional Fair
PRB	Physical Resource Block
PUSC	Partial Usage of Subchannels
QoS	Quality of Service
RR	Round Robin
RT	Real Time
SC-FDMA	Single Carrier Frequency Division Multiple Access
SIMO	Single Input Multiple Output
SISO	Single Input Single Output
SNIR	Signal to Noise plus Interference Ratio

SNR	Signal to Noise Ratio
SRP	Sequential Rectangle Placement
TDD	Time Division Duplex
TDMA	Time Division Multiple Access
TTI	Time Transmit Interval
UE	User Equipement
UL	Uplink
UMTS	Universal Mobile Telecommunication System
WCDMA	Wideband Code Division Multiple Access
WiFi	Wireless Fidelity (IEEE 802.11)
WiMAX	Worldwide Interoperability for Microwave Access (IEEE 802.16)

List of Symbols

n	Vector size
α_i	Power of the i^{th} tap
β	Orthogonality factor
γ	Pathloss exponent
σ	Thermal noise
θ_k	Direction from the base station to the user U_k
Θ_B	Half-power beamwidth
$\underline{B_k}$	Instantaneous throughput of user U_k
$\overline{B_k}$	Mean throughput of user U_k
BW	Channel Bandwidth
C_k	Channel capacity of user U_k
CS_k	OFDMA slot capacity for a user U_k
d_k	Distance between user U_k and its base station
$G_{i\,k}$	Spatial orthogonality factor between users U_i and U_k
G_0	Out-beam orthogonality factor
H_k	Channel state of user U_k
L	Number of taps
P_k	Emitted signal power of user U_k
Q_k	Queue length of user U_k
$N_{available\ slots}$	Number of OFDMA slots available for transmission
N_{BE}	Number of BE sessions
N_{BE_B}	Number of Bronze BE sessions
N_{BE_G}	Number of Gold BE sessions
N_{BE_S}	Number of Silver BE sessions
N_{bursts}	Number of bursts to be mapped
N_{GBR}	Number of GBR sessions
N_{queued}	Number of users waiting for transmission
N_{symb}	Number of symbols per scheduling period
N_{subch}	Number of subchannels/PRBs
N_{RT}	Number of RT sessions
N_{Rx}	Number of receive antennas

N_{Tx}	Number of transmit antennas
NTk	Number of OFDMA slots required by user U_k
S_k	Received signal power of user U_k
TTI	Scheduling period duration
U_k	k^{th} user

Part I

Introduction and Motivations

Chapter 1

Introduction

Telecommunication networks are more and more wide spread these days. Back in the late nineties, GPRS (General Packet Radio Service) and its evolution EDGE (Enhanced Data for GSM Evolution) started to answer the mobility desire of the data network users. The third generation of cellular networks, in operation since 2003, is called UMTS (Universal Mobile Telecommunication System). Applying to cellular networks strategies from the Internet such as packet switching and the IP protocol, UMTS and its HSPA (High Speed Packet Access) evolution are able to offer a true broadband access to Internet to mobile terminals.

Similarly, WiFi (Wireless Fidelity (IEEE 802.11)) offers domestic wireless access to a Local Area Network (WLAN), and Internet access through a DSL (Digital Subscriber Line) connection.

Moreover, fourth generation (4G) standards are now on their way, promising even faster and more robust access with lower latency. Two quite similar standards are currently competing. On one side, there is the Worldwide Interoperability for Microwave Access (IEEE 802.16) (WiMAX), which is pushed forward by the IEEE, and has already been commercially deployed. On the other side, the Third Generation Partnership Project (3GPP) proposes the Long Term Evolution (LTE), and its LTE-Advanded evolution which should be deployed soon.

1.1 Cellular networks architecture

These wireless communication standards are organized in cellular networks, which can overlap with each other. Indeed, a given area can for example be covered by Global System for Mobile Communications (GSM) or High Speed Packet Access (HSPA) and WiMAX networks while being in the range of several domestic WiFi access points. Focusing on a

Figure 1.1: Cellular network

specific cellular network technology, the covered area is divided into a grid of cells, generally of hexagonal representation, as Fig. 1.1 illustrates. Each cell is equipped with a network access point, where all of its User Equipements (UEs) are connected. These access points are called *Node B* in UMTS terminology, Base Station (BS) in WiMAX, *eNode B* in LTE, or simply Access Point (AP) in WiFi. It is the access point which shares the wireless connection between users, selecting for each Time Transmit Interval (TTI) the user, or the group of users granted for transmission. This sharing operation is called *scheduling*.

There are two transmission directions: Downlink (DL), from the access point to the user, and Uplink (UL), from the user to the access point. These two kinds of transmission can be time alternated, in Time Division Duplex (TDD) modes, or they can be performed simultaneously on different frequency bands, in Frequency Division Duplex (FDD) modes.

1.2 Scheduling policies

Packets are scheduled one after another. There are many scheduling policies, addressing different aspects of asynchronous time-varying channel quality and different QoS requirements. Scheduling is quite an important issue since an efficient allocation of the available radio resources is one of the key parameters to provide larger throughputs to an ever increasing amount of users. Schedulers should avoid wasting resources, and should allocate them to the users who would benefit from them the most, while still providing some fairness between users and while still guaranteeing

heterogeneous QoS levels. This is actually a trade-off between these contradictory objectives which has to be found, while also taking into account the solution complexity and its feedback load requirements.

A simple scheduling technique called Round Robin (RR), periodically selects each user, in a carousel kind of way. This is the most fair strategy, from a resource point of view, since each user receives the same amount of resource as often as any other user. In terms of throughput, however, users with better channel quality benefit from better throughputs. Besides, channel quality is actually time varying. It is more throughput efficient to delay users with current poor channel quality, waiting for their channel to be in better condition, and to schedule users with current good channel quality instead. This is multiuser diversity. With a sufficient amount of users, one can hope to find good channel quality users every scheduling period.

A simple multiuser diversity strategy is to schedule the user which currently has the highest channel quality, and which gets the maximum throughput from the available radio resources. This kind of technique is called *Max Throughput*. It offers the highest cell throughput, which benefits the network operators, but at the risk of never scheduling users with poor channel quality. The Proportional Fair (PF) scheduler, which we will develop later on, is able to provide a trade-off between throughput and fairness.

There are also schedulers based on some QoS criteria such as the largest weighted delay first, and its modified version [4], minimal guaranteed throughput or maximum packet delay [5, 6]. For example, the Max Weight scheduler selects the user which maximizes its throughput weighted by the size of its queue. Many other scheduling algorithms, between the most fair RR and the one offering the largest throughputs, can be found in the literature. For example, [7] overviews several Time Division Multiple Access (TDMA) and Code Division Multiple Access (CDMA) algorithms, [8] presents different Multiple Input Multiple Output (MIMO) schedulers, while [9] compares multiple algorithms when real-time and non-real-time data are mixed.

A basic scheduling algorithm selects only one user for transmission, which will be transmitting alone during the whole TTI. This approach has been proven inefficient with poor quality channels in UMTS [10], or, as we will see later on, when a single user has not enough data to transmit during the whole TTI. As a result, a multiplexing approach allowing to schedule multiple parallel streams, which we call multistream, has been suggested to improve 3G networks, and is standard with both WiMAX and LTE.

1.3 Multiple access techniques

There are two main strategies to split radio resources within current cellular networks. The first one, used by the original UMTS (Rel '99) and its HSPA evolutions is the Wideband Code Division Multiple Access (WCDMA) technique, which is a wideband evolution of CDMA. The other one, chosen by both WiMAX and LTE, is the Orthogonal Frequency Division Multiple Access (OFDMA) technique.

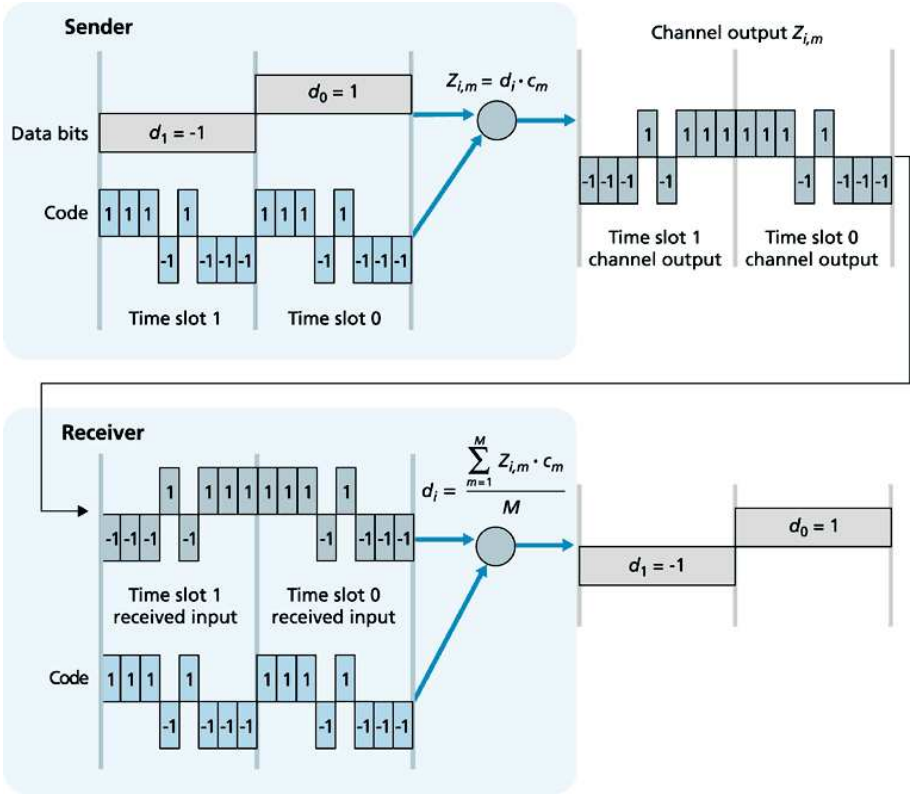


Figure 1.2: CDMA coding and decoding, taken from [1]

1.3.1 CDMA

As its name suggests, CDMA uses orthogonal codes to split radio resources. Figure 1.2 details how transmissions are encoded and decoded. In this example, the sender has the 8-chirp code $(1, 1, 1, -1, 1, -1, -1, -1)$, and wants to transmit the sequence $\{-1, 1\}$. The channel output is the multiplication of the code by -1, followed by the multiplication of the

code by 1. On the receiver side, the channel output is correlated by the code and normalized according to the size of the code to recover the transmitted sequence $\{-1, 1\}$.

When there are multiple simultaneous senders, as shown on Fig. 1.3, each transmission is multiplied by the code of its sender. The channel output is the sum of all encoded simultaneous transmissions. When it is multiplied by the code of a given user, the transmitted sequence of that particular user can be recovered, with no interference from other senders, provided that the codes are orthogonal.

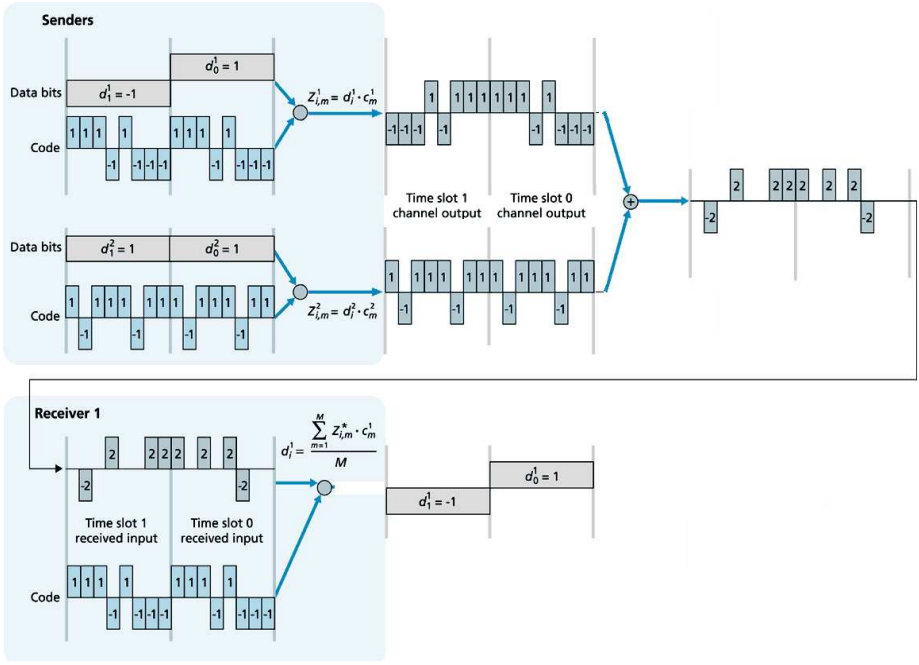


Figure 1.3: CDMA coding and decoding with multiple senders, taken from [1]

So in theory, there should be no interference from simultaneous transmissions. But actually, this requires perfect synchronization of the flows. Multiple Access Interference (MAI) occurs when flows are desynchronized. Received signals are actually composed of multiple delayed echoes of the same transmissions due to different paths. This causes Intersymbol Interference (ISI). The selection of simultaneous senders should therefore be done carefully, trying to limit these interferences.

1.3.2 OFDMA

Instead of orthogonal codes, OFDMA consider orthogonal frequency bands, called subcarriers. They are said to be orthogonal if, at the center frequency of each subcarrier, neighboring subcarriers have zero value. As represented on Fig. 1.4, where the plain blue subcarrier is sampled, at the top of its curve, the neighboring green and red dashed subcarriers have zero value.

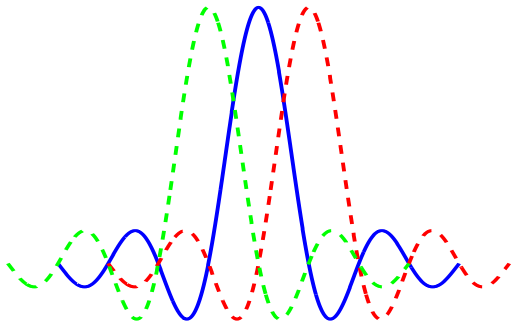


Figure 1.4: OFDM subcarrier orthogonality.

OFDMA is actually a multiple access evolution of Orthogonal Frequency Division Multiplexing (OFDM). The idea of OFDM is to divide the whole bandwidth into narrow frequency sub-bands. If frequency bands are sufficiently narrow, OFDM symbols become long enough to allow the introduction of a guard period between each symbol, filled with a copy of the last part of the following symbol, making the symbol appearing periodic. Thanks to this periodicity, and if the cyclic prefix is longer than the delay spread in the radio channel, ISI can be totally suppressed.

Time synchronization, to ensure symbol timing and correct cyclic prefix removal, and frequency synchronization, to estimate the frequency offset between transmitter and receiver, can be obtained through reference symbols, spread over subcarriers and symbols.

The multiple access evolution allows users to share these sub-bands to transmit simultaneously. Figure 1.5 compare the resource sharing to 3 users. On the left hand-side, there is the OFDM allocation, where the dark grey user benefits from all frequency sub-bands for 4 TTIs, then the light grey one for 5 TTIs and finally the white one for 3 TTIs. With OFDMA, on the center part of the figure, all three users transmit simultaneously, using different frequency bands, which can change over time.

On the right hand-side of the figure, CDMA is also portrayed. There, all users benefit from the whole frequency band during each scheduling period, and it is on another dimension, the codes, that the users are separated.

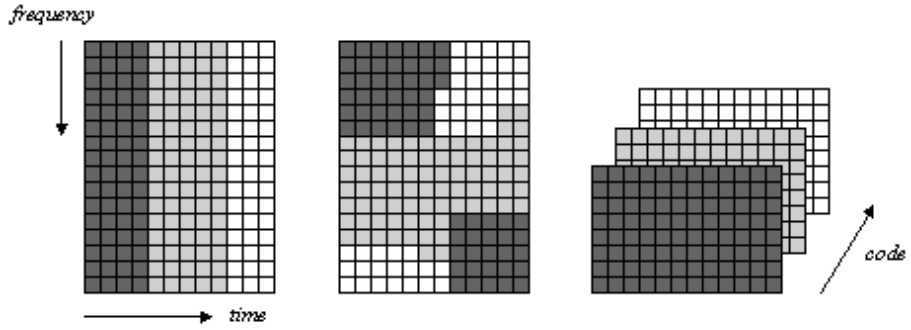


Figure 1.5: Comparison of OFDM, OFDMA and CDMA

1.4 Objectives of the Thesis

The objective of this thesis is to study scheduling, and to propose multistream enhancement heuristics, allowing to schedule multiple parallel data streams. Indeed, with a TTI scale of a few milliseconds, optimal and exhaustive computations can not be performed. Clever heuristics have to be found, in order to maximize usage of resources and to limit wastes, to improve current and future cellular networks' performance.

A fundamental CDMA issue is the fast and suboptimal identification of simultaneously transmitting user groups, with coupling due to interference. Each selected user interferes with all other transmitting users. So when granting a user for transmission, we have to ensure it is not going to cause trouble to already selected ones. Even if OFDMA should not suffer from this coupling issue, when considering multiple antenna transmissions such as *Beamforming*, transmitting users position matter. Users selection in MIMO OFDMA is therefore also a coupled problem.

Other challenges appear with OFDMA, such as frequency allocation. Whereas CDMA codes are interchangeable, each OFDMA frequency subband has its own channel response, due to frequency selectivity, and therefore has its own capacity. The scheduling algorithm should then try to allocate users their best frequency bands. There are also some constraints on the allocation shape, which is an issue of its own.

Traffic has its own constraints too, with some QoS requirements scheduling algorithms have to guarantee.

1.5 Contributions

Our contributions within this thesis cover different cellular environments. Focusing first on DL Single Input Single Output (SISO) CDMA-based networks, we demonstrate significative multistream gain with a multistream PF algorithm we implemented. We manage to decouple the users selection by handling inter-user interferences through an orthogonality factor only based on the users' channel, and obtained a linear allocation strategy.

We also adapted this algorithm to allocate beams in DL CDMA MIMO networks instead of the HSPA codes by handling inter-beam interferences with location-based orthogonality factors. Multistream gain is also obtained.

Following the industry trend to move from CDMA to OFDMA networks, we then focus on both WiMAX and LTE frameworks. Because the very nature of radio resource is different between CDMA and OFDMA, we designed a quite different scheduler, allocating resources according to some treemap visualization algorithms we tuned. It is able to provide QoS guarantees through a priority-based strategy while remaining quite simple. Considering only an average Channel Quality Indicator (CQI), it can not benefit from frequency diversity, but its feedback requirement is tremendously reduced. Compared to two reference algorithms, our computer simulations show that loosing frequency diversity to reduce feedback load is not penalizing, especially for WiMAX networks.

We have also modified this OFDMA algorithm to handle beamforming. With a Grid of Beams (GoB) strategy, we are able to seriously increase the frequency reuse factor while remaining simple and keeping feedback requirements low.

We finally focus on the issue of WiMAX burst mapping. We derived from a treemap visualization algorithm an innovative burst mapping algorithm. Compared to several reference algorithms, we obtain much better results, at the price of slightly heavier but still acceptable computations.

1.6 Publications

Here is a list of publications relative to this thesis work:

- J. Vanderpypen and L. Schumacher, Multistream Proportional Fair Scheduling Applied on Beamforming Technologies, COST 2100 TD(08) 663, October 2008.

- J. Vanderpypen and L. Schumacher, Multistream Proportional Fair Packet Scheduling Optimization in HSDPA, Proceedings of the 15th Annual Symposium of the IEEE/CVT Benelux Chapter SCVT 2008.

- J. Vanderpypen and L. Schumacher, Tiling Subchannels/PRBs in Mobile WiMAX/LTE, COST 2100 TD(09) 967, September 2009.

- J. Vanderpypen and L. Schumacher, Treemap-based Burst Mapping Algorithm for Downlink Mobile WiMAX Systems, Proceedings of the IEEE 74th Vehicular Technology Conference (VTC-Fall), September 2011.

Peer-reviewed papers can be found in Appendix A.

1.7 Structure of the document

After some introduction, this thesis is decomposed into two main parts, respectively about 3G CDMA-based UMTS in Part II, and 4G OFDMA-based WiMAX and LTE in Part III. Table 1.1 details the following structure.

Table 1.1: Thesis structure

		Part II	Part III	
		CDMA / HSPA	WiMAX	LTE
DL	SISO	Ch 3 - Multistr. PF	Ch 7 - Treemap Ch 9 - sqTM	Ch 7 - Treemap
	MISO	Ch 4 - Beamforming Ch 4 - GoB	Ch 8 - GoB	Ch 8 - GoB
UL	SISO	X	Ch 3 - Treemap	X

The CDMA part first discusses the High Speed Downlink Packet Access (HSDPA) scheduling problem in Chapter 2, then Chapter 3 presents our SISO heuristics and analyzes their performance. Multiple antenna heuristics are then investigated in Chapter 4, before drawing some conclusions and discussing the current perspectives of CDMA-based transmissions in Chapter 5.

The OFDMA part then begins with a presentation of WiMAX and LTE environments in Chapter 6. Chapter 7 presents treemap visualization, details our treemap-based scheduling heuristic, and evaluates its performance. We then introduce and evaluate the performance of a MIMO extension of this algorithm through a GoB strategy in Chapter 8. Afterwards, Chapter 9 tackles the DL WiMAX burst mapping issue, and presents a treemap based burst mapping algorithm.

Conclusions are finally drawn in Part IV.

Part II

CDMA - HSDPA

Chapter 2

The scheduling problem in CDMA-based networks

Third generation networks like the original Universal Mobile Telecommunication System (UMTS) and its HSDPA and High Speed Uplink Packet Access (HSUPA) evolutions are based on the WCDMA multiple access technique, where transmissions are encoded with orthogonal codes. As a result, we have naturally begun our investigations considering this kind of transmission mode.

2.1 The HSDPA standard

We focused our CDMA analysis on HSDPA transmissions, sometimes branded as 3.5G, or 3G+. HSDPA consists of three DL channels, and one UL channel [11]. The DL data transmissions happen on the HS-DSCH (High Speed Dedicated Shared Channel), other channels are for control information and feedback about transmission.

The HS-DSCH is divided into 15 physical channels determined by orthogonal channelization codes. They are shared between active users by the scheduler. Originally, the standard prescribes that for each cell, only a single user is granted for transmission per TTI, receiving all the 15 codes. Actually, [10] has shown that this approach is suboptimal. When channel quality is poor, when a user has not enough data to transmit, or when its UE can not handle 15 codes, some of them are actually wasted. It would be cleverer to let several users share these codes, and transmit simultaneously. As [12] points out, the selection of these simultaneous users should be done carefully, to limit interference between each other.

Moreover, scheduling, and therefore user selection, has to be done at TTI scale, which is only 2 ms [13] in HSDPA. This is the reason why we looked for an heuristic.

2.2 Proportional Fair Scheduling

There are different scheduling policies. As we already mentioned, among other schedulers, there are RR, which is quite fair but does not benefit from multiuser diversity, or the Maximum Throughput algorithm, which provides high throughputs with no fairness guarantee. In between, PF takes advantage of multiuser diversity while allowing an appreciable trade-off between user fairness and cell throughput.

Instead of simply maximizing the potential throughput of the users, PF weighs throughputs by the backlog of the connection, such as the maximizing PF_k criteria is

$$PF_k = \frac{B_k}{\overline{B_k}} \tag{2.1}$$

where B_k is the instantaneous throughput of the user U_k, and $\overline{B_k}$ is its mean throughput. The task of the PF scheduling algorithm is to find the user U_{k^*} who maximizes Eq. (2.1) As a result, even if the instantaneous throughput is considered, the mean throughput weight protects poor channel quality users from starvation. Indeed, if a user close to the base station has constantly a good channel and data to transmit, he will initially be frequently scheduled. But after some time, his mean throughput will increase, and his ratio in Eq. (2.1) will eventually get lower than the one of another user, with poor channel quality, which has not transmitted yet.

The mean throughput at a given TTI t_t is obtained from a moving average, depending on a forgetting parameter $\alpha \in [0, 1]$:

$$\overline{B_k}(t = t_t) = \alpha \cdot \overline{B_k}(t = t_{t-1}) \ + \ (1 - \alpha) \cdot B_k(t = t_k) \tag{2.2}$$

The potential throughput of each user, given its channel quality, has therefore to be known. It is actually upper bounded by the channel capacity C_k, which depends on the Signal to Noise plus Interference Ratio (SNIR), according to the Shannon theorem [14]:

$$C_k = BW \cdot \log_2(1 + SNIR_k) \tag{2.3}$$

where BW is the channel bandwidth. Shannon theorem actually provides an overestimated bound, considering ideal modulation and coding scheme.

The SNIR is the ratio in between the signal power and the sum of the noise and the interference powers:

$$SNIR \ = \ \frac{Signal}{Noise \ + \ \underbrace{ISI + MAI_{intra} + MAI_{inter}}_{Interferences}} \tag{2.4}$$

The interferences are divided among several categories. The ISI are due to multiple delayed echoes of the original transmissions. Indeed, in real environment, transmitted signal undergoes numerous reflexions, refractions and other attenuations. It can therefore reach destination through different alternative paths of various length. So multiple delayed echoes are received. These interferences are caused to users by their own transmissions, while interferences due to other users' transmissions are the MAI. They are of two kinds, the MAI_{intra}, due to users of the same cell, and the MAI_{inter}, due to users of other cells.

The ISI can be estimated by the orthogonality factor β as modeled by [15], giving the proportion of the transmit power which is perceived as interference.

$$\beta = 1 \; - \; \frac{\sum_{i=1}^{L} (\alpha_i)^4}{\left(\sum_{i=1}^{L} (\alpha_i)^2 \right)^2} \tag{2.5}$$

where the $\{\alpha_i\}_{i=1:L}$ are the powers of the L taps (most significant echoes) modeling the channel dispersion of the transmitted signal. One can easily see that if there is only one significant tap, we get $\beta \approx 0$, and there is no ISI. On the contrary, if power is equally distributed among all taps, the interferences are maximum.

Therefore, ISI undergone by user u_k depends on its emitted power P_k, on its channel state $|H_k|$, and on its pathloss d_k^{γ}, which represents the signal attenuation over a distance d_k according to a pathloss exponent γ, as follows:

$$ISI_k = \beta_k \cdot |H_k|^2 \cdot \frac{P_k}{d_k^{\gamma}} \tag{2.6}$$

If CDMA codes are not reused by different users within a cell, the only interfering effect of other users' transmission is caused by reception of delayed echoes of other users' symbols. MAI_{intra} are therefore also a kind of ISI, and can also be estimated with the same β factor giving the proportion of other users transmit power perceived as MAI_{intra}:

$$MAI_{intra_k} = \beta_k \cdot |H_k|^2 \cdot \frac{\sum_{u \neq k} P_u}{d_k^{\gamma}} \tag{2.7}$$

As far as modeling MAI_{inter}, we considered random values obtained through G-factor. The G-factor is the ratio between intra- and inter-cell interference, as [16, p. 86] defines:

$$G_{factor} = \frac{ISI + MAI_{intra}}{MAI_{inter}} \tag{2.8}$$

Its probability distribution in a macrocell scenario is shown in [16, Fig. 5.7]. So drawing random G-factor values, we can estimate MAI_{inter}

given $ISI + MAI_{intra}$:

$$MAI_{inter} = \frac{ISI + MAI_{intra}}{G_{factor}} \quad (2.9)$$

$$= \frac{1}{G_{factor}} \cdot \left(\beta_k \cdot |H_k|^2 \cdot \frac{\sum_u P_u}{d_k^\gamma} \right) \quad (2.10)$$

As far as the received signal power S_k is concerned, it can be derived from:

$$S_k = |H_k|^2 \cdot \frac{P_k}{d_k^\gamma} \quad (2.11)$$

With σ standing for the noise, the SNIR can therefore be obtained from

$$SNIR_k = \frac{|H_k|^2 \cdot \frac{P_k}{d_k^\gamma}}{\sigma + \left(1 + \frac{1}{G_{factor}}\right) \cdot \beta_k \cdot |H_k|^2 \cdot \frac{\sum_u P_u}{d_k^\gamma}} \quad (2.12)$$

2.3 Multistream PF scheduler and coupling issues

As we already mentioned, scheduling only a single user per TTI is inefficient. It probably can not benefit from all resources, which are the 15 HSDPA codes, especially if it has a bad quality channel, or if it does not have a lot of data to transmit.

The multistream scheduling problem is therefore the selection of the best K users among the N_{queued} waiting ones:

$$(users^*, resources^*) = \underset{\substack{users \in C_K^{N_{queued}} \\ resources}}{\arg\max} \sum_{k=1}^{K} \frac{B_k(resources)}{\overline{B_k}} \quad (2.13)$$

where the potential throughputs B_k actually depend on the resource allocation we perform, and $users$ is the users combination we select for transmission.

This problem is rather complex. For a given K value, there are $C_{N_{queued}}^K$ possibilities of users selection. And once the K users are known, the resources still have to be allocated. Considering the 15 codes can be allocated one by one to those K users, it is actually a combination with repetition, and there are C_{15-1}^{15-K} more possibilities. It corresponds to the allocation of $15 - K$ codes to the K users. Indeed, only $15 - K$ codes

have to be allocated since each one of the K users must have at least one code, or there are not K simultaneous users. As a result, an exhaustive research implies

$$\sum_{K=1}^{15} C_{N_{queued}}^{K} \cdot C_{15-1}^{15-K} \tag{2.14}$$

potential solutions to evaluate. For example, with only 8 users, it means to evaluate 170,544 potential solutions.

2.4 Scheduling as a maximum clique search problem

In the aim to improve the group selection of the simultaneously transmitting users, we looked for a heuristic suiting the problem. In [17], it was proposed to model the users selection as a clique search problem.

In graph theory, a clique is a set of nodes where each of them is linked by a vertex to every other node of the clique. A full mesh network can be represented by a clique. The idea is here to define a quasi-orthogonality relation between users, depending on whether two users would significantly interfere with each other's transmission or not. If two users do not interfere too much with each other, they are considered quasi-orthogonal, and they can be scheduled simultaneously. As far as scheduling is concerned, the idea would be to build cliques of quasi-orthogonal users, and the strategy to limit MAI would be to pick up all simultaneous users from the same clique.

This quasi-orthogonality relation seemed quite interesting since we managed earlier to model interferences with orthogonality factors. But actually, building up such cliques is quite a hard job, since this is a NP-complete problem [18]. Exact algorithms looking for a maximum clique [19], such as a point removal algorithm, or backtracking and branch and bound strategies, have a time complexity of $O(nm\mu)$, where n, m, and μ are the number of vertices, edges and maximal clique size of a graph. Moreover, sequential greedy or local search heuristics seem to have a high time complexity. There are also some other algorithms, based on neural networks, genetic algorithms, tabu search or simulated annealing, but they are also too complex to provide us with a solution for our scheduling problem within a TTI of a few milliseconds.

Moreover, these algorithms usually provide the largest cliques, but in scheduling scenarios, we only have a limited amount of orthogonal codes to allocate. We are not interested by cliques of sizes larger than

the amount of codes we have. Once cliques are built, another algorithm should be run to pick the optimal clique to work, and then its best users. As a result, even if this strategy has a strong mathematical background, it seems to be heavier to run than some simple heuristics we are going to present in the next chapter, and does not fully answer the scheduling problem, since the selection of a clique, and the selection of users within a clique remain open problems.

Chapter 3

HSDPA SISO heuristics

A solution to the problem of Eq. (2.13) proposed in [20] is first to select users without taking care of interferences, so this selection can be done linearly. Then on a second time, to allocate codes and transmit powers to users trying to limit interferences.

Actually, interferences can rather be simplified. Considering scheduling at cell level, user selection and resource allocation can not be based on MAI_{inter}. And as far as MAI_{intra} is concerned, since HSDPA is about DL scenario, there is no synchronization issue between flows. Simultaneous flows are emitted synchronously by the base station, and each user receives all flows at the same time, with the same multipath effect on all of them. The only MAI_{intra} is due to delayed taps, just like ISI. The difference between ISI and MAI_{intra} is that ISI are caused by multiple echoes of your own transmissions, whereas MAI_{intra} are caused by multiple echoes of other users' transmissions.

As a result, the intracell interference in DL HSDPA only depends on the total transmitted power and on the tap distribution, and not on the selection of any particular user. The simplification of ignoring MAI_{intra} when selecting simultaneous users as suggested by [20] does therefore not degrade scheduling performance in HSDPA. As far as MAI_{inter} are concerned, they are still modeled through the G-factor.

The SNIR of a given user U_k can be obtained from

$$SNIR_k = \frac{|H_k|^2 \cdot \frac{P_k}{d_k^\gamma}}{\sigma + \beta_k \cdot |H_k|^2 \cdot \frac{\sum_u P_u}{d_k^\gamma} \cdot \left(1 + \frac{1}{G_{factor}}\right)} \tag{3.1}$$

without paying any attention to who are the other simultaneous users.

Let us point out that in UL scenarios, there is no synchronization between flows, and therefore the MAI can not be simply modeled by the orthogonal factor β. Distances between users intervene, and the selection of users becomes much more complicated.

3.1 Users selection

The users selection is a multi objective problem. To provide a good trade-off between throughput and fairness, we consider as criteria to maximize the PF. Indeed, it benefits from multiuser diversity, selecting users when their channel is good, while still providing some fairness between users through the backlog weights. As far as performance evaluation of the algorithms is concerned, we are going to compare algorithms both on throughput and fairness aspects, to assess the trade-off we obtained.

Since the user selection does not influence MAI_{intra}, users can be chosen independently one after another, only according to their PF criteria. First, we have to estimate the potential throughput of each user. This can be derived from the channel capacity, whose estimation can be performed through Shannon's theorem (Eq. (2.3)). We thought channel capacity should be truncated depending on user's queue. Indeed, even with a very good channel quality, if a user has only limited data to transmit, he will not have a high throughput. As a result, we derive the potential throughput B_k of a single code from

$$B_k = \min\left\{ BW \cdot \log_2\left(1 + SNIR_k\right),\ \frac{Q_k}{TTI} \right\} \tag{3.2}$$

where Q_k is the queue length of the user U_k, and TTI is the TTI duration.

Once potential throughputs are known, we can sort users according to their PF criteria:

$$\left\{ U_{[1]},\ U_{[2]},\ \ldots,\ U_{[N_{queued}]} \ \text{ such as } \ \frac{B_{[k]}}{\overline{B_{[k]}}} \geq \frac{B_{[k+1]}}{\overline{B_{[k+1]}}} \right\} \tag{3.3}$$

Then, we have to perform the resource allocation. There are actually two kinds of resources. The HSDPA codes we already mentioned, but also the transmit powers. While power is a continuous resource, codes can not be half allocated.

Hereafter, we will discuss about two allocation strategies.

1. We can equally spread the power among the HSDPA codes, and then allocate these codes to users.

2. We can instead equally share the power between transmitting users, then let them choose how many codes they want to use, paying attention to the 15-code limit of the cell.

3.2 Power equally shared among codes

A first allocation strategy is to equally share the available power between codes. Considering all 15 codes are used, we can estimate the throughput each user would obtain from one code with $\frac{1}{15}$ of the HSDPA transmit power.

A basic code allocation would be to give all codes to a single user, like traditional scheduling, and then to compare the throughput improvement if a fraction of the codes is allocated to another user. As a next step, we check whether it would be better to allocate another fraction of the codes to a third user, and so on.

Given its UE, a user can only receive 5, 10 or 15 codes [21, p. 368]. We could therefore compare the allocation of

- 15 codes to a single user;
- 10 codes to one user, and 5 to another one;
- 5 codes to three users.

Actually, an exhaustive research for the best user, then for the best pair of users, and then for the best triplet is of $\mathcal{O}(N_{queued}^3)$. But since we have shown that interference does not depend on who are the transmitting users, we can just sort users according to their PF criteria as Eq. (3.3) shows, and then compare the allocation of 15 codes to the best user vs the allocation of 10 codes to the best user, and 5 codes to the second best vs 5 codes to the three best users. The procedure is synthesized in Algorithm 1.

Moreover, if we consider that users can handle codes one by one, and not by groups of five, we can improve the algorithm. We can allocate to a first user as many codes as he needs, and then, if some codes remain unused, they can be allocated to other users. And because interference does not depend on which user is going to use each code, accurate throughput estimations can be done, even when selecting the first user. This assumes of course that all codes will eventually be allocated so the total transmit power will be used.

In this other version, synthesized by Algorithm 2, the first step is still to evaluate the throughput each user can obtain from one code. Then the amount of codes each user requires and the PF criteria associated are evaluated. The user k^* that maximizes its PF criteria receives the NC_{k^*} codes he requires, and the remaining users are tested again to identify the one which would benefit the most from the remaining codes, and so on.

Algorithm 1 Multistream HSDPA PF scheduler, codes allocated 5 by 5

Considering a total transmit power P_{HSDPA},

for all users U_k **do**

Evaluate SNIR:

$$SNIR_k = \frac{|H_k|^2 \cdot \frac{P_{HSDPA}/15}{d_k^\gamma}}{\sigma + \beta_k \cdot |H_k|^2 \cdot \frac{P_{HSDPA}}{d_k^\gamma} \cdot \left(1 + \frac{1}{G_{factor}}\right)}$$

Evaluate potential throughput of one code:

$$B_k = \min\left\{BW \cdot \log_2\left(1 + SNIR_k\right),\ \frac{Q_k}{TTI}\right\}$$

end for

Sort users according to their PF criteria:

$$\left\{U_{[1]},\ U_{[2]},\ \ldots,\ U_{[N_{queued}]} \text{ such as } \frac{B_{[k]}}{\overline{B_{[k]}}} \geq \frac{B_{[k+1]}}{\overline{B_{[k+1]}}}\right\}$$

Then evaluate

- $PF_{1\ user} = \frac{\min\left\{15 \cdot B_{[1]},\ \frac{Q_{[1]}}{TTI}\right\}}{\overline{B_{[1]}}}$
- $PF_{2\ users} = \frac{\min\left\{10 \cdot B_{[1]},\ \frac{Q_{[1]}}{TTI}\right\}}{\overline{B_{[1]}}} + \frac{\min\left\{5 \cdot B_{[2]},\ \frac{Q_{[2]}}{TTI}\right\}}{\overline{B_{[2]}}}$
- $PF_{3\ users} = \frac{\min\left\{5 \cdot B_{[1]},\ \frac{Q_{[1]}}{TTI}\right\}}{\overline{B_{[1]}}} + \frac{\min\left\{5 \cdot B_{[2]},\ \frac{Q_{[2]}}{TTI}\right\}}{\overline{B_{[2]}}} + \frac{\min\left\{5 \cdot B_{[3]},\ \frac{Q_{[3]}}{TTI}\right\}}{\overline{B_{[3]}}}$

and select the highest PF solution.

Algorithm 2 Multistream HSDPA PF scheduler, codes allocated 1 by 1

Considering a total transmit power P_{HSDPA},
for all users U_k **do**
Evaluate SNIR:

$$SNIR_k = \frac{|H_k|^2 \cdot \frac{P_{HSDPA}/15}{d_k^\gamma}}{\sigma + \beta_k \cdot |H_k|^2 \cdot \frac{P_{HSDPA}}{d_k^\gamma}}$$

Evaluate potential throughput of one code:

$$B_k = \min\left\{ BW \cdot \log_2\left(1 + SNIR_k\right),\ \frac{Q_k}{TTI} \right\}$$

end for
while there are still unused codes and users to be scheduled **do**
for all users U_k **do**
Find the desired number of codes NC_k for each user yet to schedule, with NC_{unused} standing for the number of yet unallocated codes:

$$NC_k = \underset{N \le NC_{unused}}{\arg\max} \left\{ N \cdot B_k \ \text{ such as } N \cdot B_k \le Q_k \right\}$$

Evaluate the PF criteria of each allocation: $PF_k = \frac{NC_k \cdot B_k}{\overline{B_k}}$
end for
Allocate to the user k^* the NC_{k^*} he desires, where

$$NC_{k^*} = \min\left\{ \arg\max_k \{PF_k\}\ ,\ NC_{unused} \right\}$$

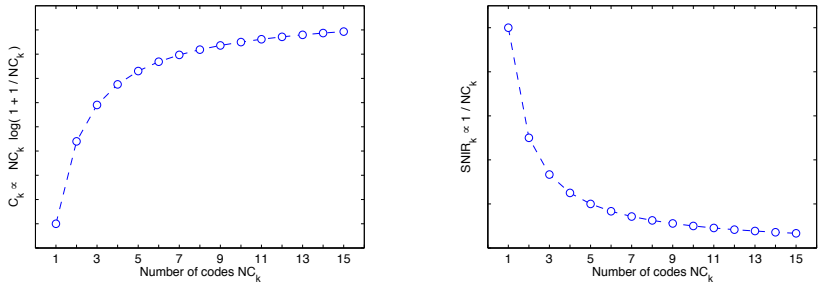
end while

Let us point out that this algorithm can lead to schedule up to 15 simultaneous users. Actually, there are some constraints on control channels. Indeed, each simultaneous user require at least one High-Speed Shared Control Channel (HS-SCCH) to demodulate its informations transferred through the HSDPA channel. HS-SCCH uses a spreading factor (SF) 128, but since HSDPA already uses 15 SF 16 channels, there can only be 8 HS-SCCH, therefore limiting in practice the number of simultaneous users to 8.

3.3 Power shared over users

Instead of sharing out the power to the codes, we could share it directly among the users, and let them decide to how many codes they want to spread the power they received.

As Fig. 3.1(a) shows, for a given transmit power, the more the power is spread on many codes, then the more the total Shannon's theoretical capacity $C_k \propto NC_k \cdot \log(1 + \frac{1}{NC_k})$ is high. However, the more codes we use, then the less power each code-dependant channel receives. As a result, its $SNIR_k \propto \frac{1}{NC_k}$ diminishes, as Fig. 3.1(b) shows.



(a) Effect on total capacity (b) Effect on the SNIR of one channel

Figure 3.1: Effects of the power repartition over multiple channels

So even if using a lot of channels provides a better theoretical total throughput, it reduces the quality of each of the multiple channels. Here on Fig. 3.1 we consider the Shannon bound on channel capacity, but actually, channel capacity depends on channel coding. Capacity curve is shaped like a stairway, where each step corresponds to a particular coding scheme, which use requires a minimum level of SNIR. A too low SNIR leads to use poor coding schemes, causing poor throughputs, whereas a little more power would have allowed the use of a more efficient coding scheme. Therefore there is a trade-off to find to use a maximum amount of channels while preserving a SNIR sufficiently high for each of these channels.

The idea is first to find the user maximizing its PF criteria, considering it has been allocated the full transmit power. Then we allocate it as many codes as it seems optimal. If there remains free codes, we start looking for a second user, which would benefit the most from these free codes, considering it receives half of the power of the first user. Afterwards, if there are still available codes, we try to find a third user which would maximize its PF criteria, considering it receives a third of the full power, and so on. Once all codes have been allocated, all throughputs must be recomputed, since the power allocation varies until all codes have been allocated. This procedure is summarized in Algorithm 3.

Algorithm 3 Multistream HSDPA PF scheduler, power spread over users

while there are still unallocated codes **do**
  for all unscheduled users U_k **do**
    Considering N_{sched} is the number of already scheduled users, set the transmit power of yet unscheduled user U_k:

$$P_k = \frac{P_{HSDPA}}{N_{sched} + 1}$$

    Find the required amount of codes NC_k:

$$NC_k = \underset{NC \leq NC_{unused}}{\arg\max} \left\{ NC \text{ such as } NC \cdot B_k(NC) \leq Q_k \right\}$$

    where the throughput $B_k(NC)$ actually depends on the number NC of codes the power P_k is spread on:

$$B_k(NC) = BW \cdot \log_2 \left(1 + \frac{|H_k|^2 \frac{P_k}{NC} \frac{1}{d_k^\gamma}}{\sigma + |H_k|^2 \beta_k P_{HSDPA} \frac{1}{d_k^\gamma} \cdot \left(1 + \frac{1}{G_{factor}} \right)} \right)$$

    Derive its PF criteria: $PF_k = \frac{NC_k \cdot B_k(NC_k)}{\overline{B_k}}$
  end for
  The user U_{k^*} receives its requires NC_{k^*} codes, with $k^* = \arg\max_k PF_k$
end while
Re-evaluate all throughputs, with actual transmit power:
for all scheduled users U_k **do**

$$B_k = NC_k \cdot BW \cdot \log_2 \left(1 + \frac{|H_k|^2 \frac{P_{HSDPA}}{N_{sched}} \frac{1}{d_k^\gamma}}{\sigma + |H_k|^2 \beta_k P_{HSDPA} \frac{1}{d_k^\gamma} \cdot \left(1 + \frac{1}{G_{factor}} \right)} \right)$$

end for

A drawback of this strategy is that when we allocate the codes, we still do not know how much power the users will receive. The choices are based on a fraction of power they may not receive. This kind of inverse water-filling approach does not seem ideal, even if it allows users to decide on how many codes they want to share out their power, which can be useful to reach a given SNIR threshold, allowing the use of a coding scheme leading to a better throughput. Indeed, if a user reduces its number of codes to ensure a given SNIR threshold, and then its allocated power is reduced, its SNIR threshold is not achieved.

An idea to improve this scheme would be to decide first how many simultaneous users we would accept, so the power allocation will be correct the first time. However, if the first users exhaust all the codes, and less users than foreseen can be scheduled, some power is leftover. This leftover power could then be shared between all scheduled users, to improve their channels quality. Another idea could also be to limit the number of codes a user can receive, to ensure we can schedule as many users as foreseen.

One more improvement would be to use a water filling algorithm to allocate power to users, instead of giving the same amount of power to everyone, but that would not solve the issue of code selection based on fractions of power users might not receive.

3.4 Performance evaluation

From early tests, it appears Algorithms 1 and 3 globally have the same performances, while Algorithm 2 produces better results. Actually, since the DL interferences do not depend on which users are selected, allocating codes by groups of 5, as Algorithm 1 does, is not necessary. And as far as Algorithm 3 is concerned, allocations based on power users would probably not have, does not seem really efficient. This section therefore presents throughput and fairness performance of Algorithm 2, equally spreading the power on codes, and allocating them one by one. This algorithm is called Multistream PF.

3.4.1 Some reference algorithms

The Multistream PF algorithm has been scripted in MATLAB, and compared to four classical single user schedulers: RR, Max Throughput, classical PF and Max Weight scheduler. We also implemented a multistream RR, to distinguish the gain of our algorithm from the multistream gain, and benchmarked results against exhaustive search results.

Single stream RR selects each user at a periodic time slot. Multistream RR also selects users in a carrousel kind of way, but when a selected user can not benefit from all codes, the unused ones are allocated to the following users.

Max Throughput allocates all codes to the user k_{MT} which maximizes its throughput:

$$k_{MT} = \arg\max_k \left\{ \min\left\{ C_k, \frac{Q_k}{TTI} \right\} \right\} \tag{3.4}$$

Single stream PF allocates all codes to the user k_{PF} which maximizes

$$k_{PF} = \arg\max_k \left\{ \frac{\min\left\{ C_k, \frac{Q_k}{TTI} \right\}}{\overline{B_k}} \right\} \tag{3.5}$$

And Max Weight allocates the codes to the user k_{MW} which maximizes

$$k_{MW} = \arg\max_k \left\{ Q_k \cdot \min\left\{ C_k, \frac{Q_k}{TTI} \right\} \right\} \tag{3.6}$$

As far as the exhaustive search is concerned, though not affordable at TTI scale, those results can be used as benchmark for the different algorithms. We considered codes can be allocated one by one, and tried to maximize the sum of each user PF criteria. We evaluated all possible allocations of the 15 codes to the users, which means 15 codes for one user, or 14 codes to one user and 1 code to another one, or 13 codes to an user and 2 codes to another one, or one code to 2 other users, and so on.

As a matter of fair comparison, as one can see, all schedulers rely on throughputs truncated according to users' queues.

3.4.2 MATLAB simulation parameters

The algorithms have been scripted in MATLAB language. We model 8 users with continuous transmit demand and infinite backlog. Every TTI, the size of each user's queue is increased with some random amount of bits. To model the queue rates, we considered a fixed arrival packet rate of 1 packet per TTI, of size uniformly distributed as Table 3.1 mentions.

Channel parameters have been obtained through the SCME channel model [22], implemented by the IST-WINNER project, according to the 3GPP standard. Users have been simulated in a cell in between 35 and

Mean user queue rate	Packet sizes uniformly distributed within
0.5 Mbps	[0.5 , 1.5] kb
1 Mbps	[1 , 3] kb
2 Mbps	[2 , 6] kb
4 Mbps	[4 , 12] kb
8 Mbps	[8 , 24] kb

Table 3.1: Traffic generation model.

500m away from the access point, moving at 30kmph. Total bandwidth is 5 MHz. As [21, Table 12.7] mentions, the access point has a transmit power of 40W, whose 80% are granted for the 15 HSDPA codes. Noise power is -101.2 dBm, and TTI lengths 2ms. All parameters are summarized by Table 3.2.

Parameter	Value
Number of users	8 users
Traffic model	Fixed packet arrival rate, packets of uniformly distributed size
Channel model	SCME
Bandwidth	5MHz
Carrier Frequency	2GHz
Resource to share	15 HDSPA codes
User location	Between 35 and 500m away from access point
User velocity	30kmph
Transmit power	32W
Noise power	-101.2 dBm
Interference model	ISI + Intercell interference estimated through G-factor
Antenna pattern	SISO
TTI duration	2ms
Run duration	100 TTIs
Number of runs	400 runs

Table 3.2: Parameters for CDMA SISO computations.

3.4.3 Throughput results

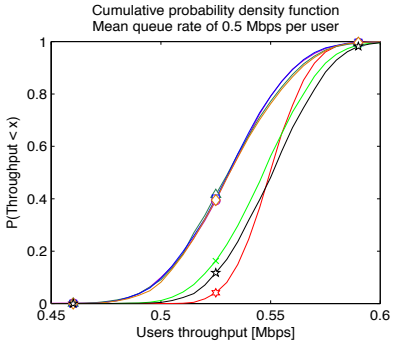
We compare cumulative probability density function of users' throughput for the different scheduling algorithms first. These results are presented at Fig. 3.2 considering different user mean queue rates, and are averaged over 400 runs of 100 TTIs each. These averaged results seem to us statistically significant, since for some algorithms, simulations were performed several times, with consistent results.

As one can see on Figs. 3.2(a) and 3.2(b), with queue rates lower than 1 Mbps, all single stream schedulers have the same throughputs, whereas multistream algorithms achieve some significative gain. With only 0.5 Mbps of mean user queue rates, both Multistreams PF and RR achieve a 12% gain at 50% outage vs single stream algorithms. This can be explained by the queue rates. Since they are low, a single user has not enough data to benefit from all codes. Multistream algorithms, allowing multiple users to transmit simultaneously, provide higher throughputs. And taking into account channel quality, Multistream PF produces slightly better results than Multistream RR, especially with queue rates larger than 1 Mbps.

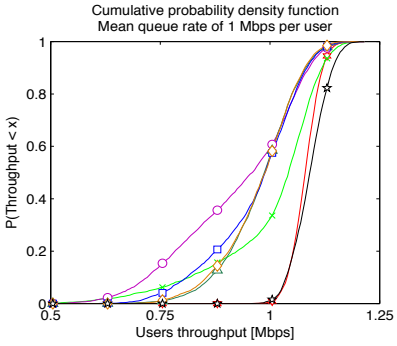
Increasing queue rates, some throughput differences appear between single stream algorithms, as Fig. 3.2(c) shows. Max Throughput and Max Weight equally produce the highest single stream throughput, slightly better than RR and PF. Multistream PF still provides the highest throughput, with around 6% gain at 50% outage on the other algorithms, but Multistream RR performance are not as good, only matching those of single stream algorithms.

Then as Figs. 3.2(d) and 3.2(e) show, when queue rates are really high, single stream PF and both RR algorithms all lead to the same throughput, while Multistream PF provides significative throughput gain. As far as Max Throughput and Max Weight are concerned, taking advantage of channel quality, they provide poorer throughputs to some users, but better throughputs to others. This lower fairness is reflected by less vertical curves, and is confirmed by the analysis of section 3.4.5.

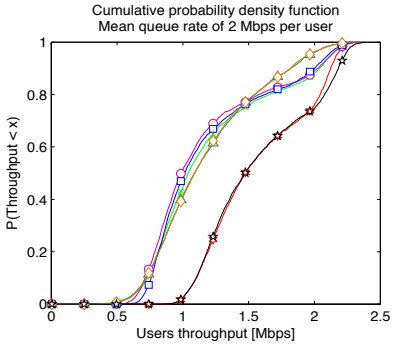
As far as an exhaustive search trying to maximize a PF criteria is concerned, throughput results are quite close to those obtained by our heuristic, as Fig. 3.2 shows. Actually, given the interference model, an exhaustive search does not provide any throughput gain compared to our heuristic. Depending on the queue rate, our algorithm can even provide better results than the exhaustive search. Actually, while the exhaustive search tries to provide the best overall throughput vs. fairness tradeoff, our algorithm allows to the user it has selected, according to a PF criteria, to maximize its own throughput.



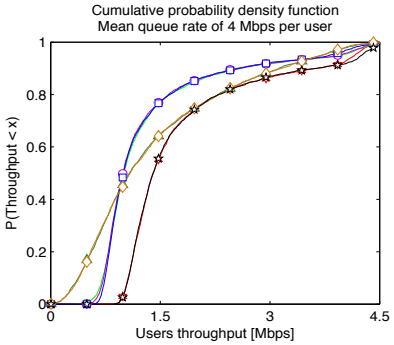
(a) 0.5 Mbps



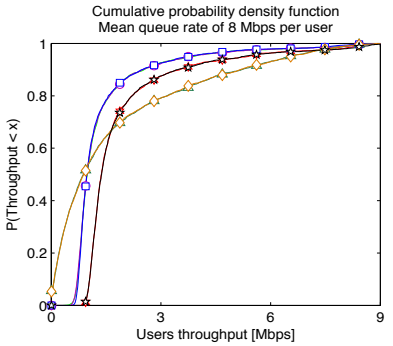
(b) 1 Mbps



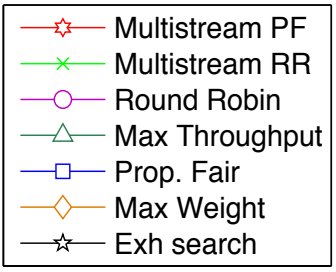
(c) 2 Mbps



(d) 4 Mbps



(e) 8 Mbps



(f) Legend

Figure 3.2: Cumulative probability density function of the users throughput, for 8 users with a mean queue rate of respectively 0.5, 1, 2, 4 and 8 Mbps.

3.4.4 Number of simultaneous users

Multistream schedulers allow several users to transmit simultaneously, but a single user can also receive all codes if he can benefit from all of them. Figure 3.3 presents the distribution of the number of simultaneously scheduled users for Multistream PF and RR as well as for the exhaustive search.

With the highest queue rates, most of the time actually only one single user is scheduled per TTI. But with lower queue rates, multistream allows several users to transmit simultaneously. All users are scheduled simultaneously around 80% of the TTIs by our algorithm when they only have a mean queue rate of 0.5 Mbps. Multistream RR and the exhaustive search select a bit more simultaneous users than Multistream PF does, as Figs. 3.3(b) and 3.3(c) shows. When considering the largest queue rates, most of the time, only one user is scheduled per TTI, the exhaustive search only allowing a bit more multistream transmissions. Indeed, for both Multistream PF and RR, when a user is selected, it can benefit from as many codes as it wants, then leftovers are allocated to other users, while the exhaustive search can decide to schedule more simultaneous users to improve the total PF criteria, even if it prevents any of them from emptying their queue.

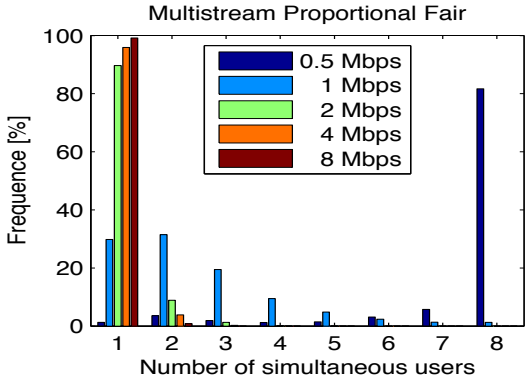
So our throughput results can be explained by the number of simultaneous users. At low queue rates, multistream algorithms allow all users to transmit simultaneously, instead of waiting for their turn, therefore producing higher throughputs. And when queue rates are getting high, differences between multistream and single stream algorithms disappear because most of the time, multistream algorithms only schedule a single user at the time.

3.4.5 Fairness results

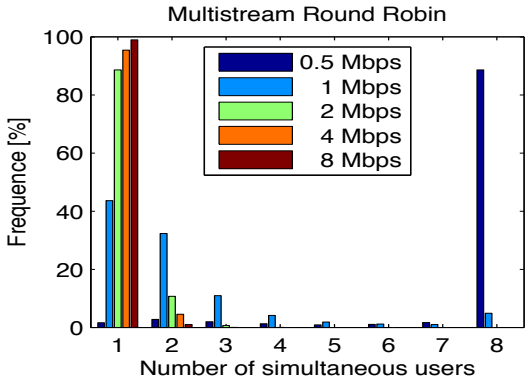
We also compared the fairness of the algorithms through the Jain Index [23]. It has the advantage to be bound between $1/N_{queued}$ and 1, to be dimensionless, and to be independent of the number of users. If we write $\tilde{B}_k$ the actual transferred throughput of the user k at the end of a simulation run, the Jain Index is defined as

$$\text{Jain Index} \equiv \frac{\left(\sum_{k=1}^{N_{queued}} \tilde{B}_k\right)^2}{N_{queued} \cdot \sum_{k=1}^{N_{queued}} (\tilde{B}_k)^2}. \tag{3.7}$$

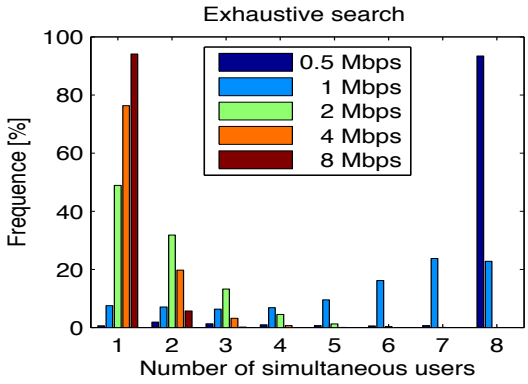
The closer to 1 the Jain Index is, the fairer the algorithm is. Indeed, if all transferred throughputs are equal, the Jain Index is equal to 1.



(a) Multistream PF



(b) Multistream RR



(c) Multistream Exh

Figure 3.3: Distribution of the number of simultaneous users for the different queue rates with multistream algoritms.

Figure 3.4 presents the Jain Index of the different schedulers, averaged on 400 runs of 100 TTIs. At low queue rates, all schedulers are equally fair, except the Multistream PF, which is a bit fairer, even able to match exhaustive search results. Increasing queue rates reduces fairness of all algorithms, but especially of Max Throughput and Max Weight.

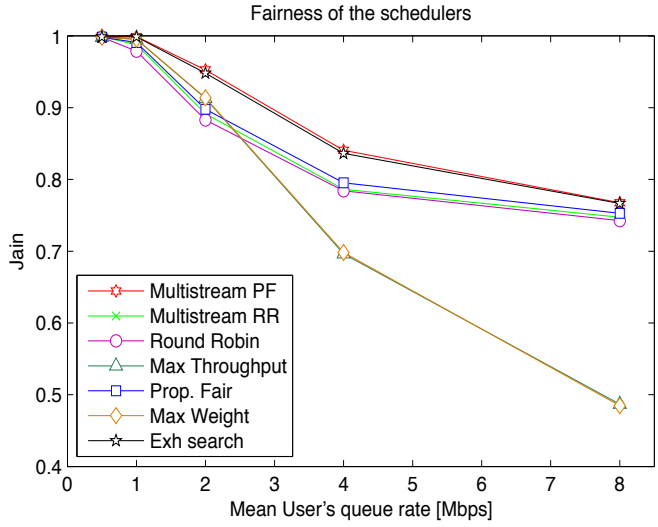


Figure 3.4: Fairness of the different schedulers, depending on the users' mean queue rate.

Indeed, when queue rates are low, all schedulers manage to fully serve all users, which all have the same queue rate. All algorithms are therefore really fair. But when queue rates are increased, not all traffic can be scheduled, and some choices have to be done. Obviously, schedulers trying to maximize throughput will not be as fair as algorithms taking into account connection backlog such as PF, or carrousel turns such as RR.

Our Multistream PF appears to be the most fair algorithm, matching the fairness of the exhaustive search. Considering current channel capacity and users needs, it is able to select more fairly users than a simple carrousel independent of these considerations. Moreover, being able to schedule multiple users simultaneously, especially at low queue rates, users can be granted for transmission more often, such that during a short term period all users can be scheduled once, therefore improving short-term fairness.

3.4.6 Performances summary

As we have seen, the multistream scheduling algorithm we set up based on a PF strategy improves both throughput and fairness performance w.r.t. traditional single stream schedulers. Moreover, our heuristic technique is able to match both throughput and fairness results of an PF-based exhaustive search.

Multistream throughput gains are more important at low to moderate queue rates. Actually with high queue rates, multistream algorithms only schedule one user per TTI, since a single user can benefit from all codes. As [24] has shown, the maximal channel capacity is obtained when only a single user is transmitting over the entire bandwidth, so it is when users' throughput is truncated by their queue length and not by their channel capacity that multistream gain can be found.

So while the authors of [10] pointed out the throughput inefficiency of single stream transmissions when channel quality is poor, one can point out from this work that when queue rates are too small, multistream scheduling is much more efficient than single stream scheduling.

As far as fairness is concerned, our Multistream PF algorithm is fairer than any other implemented scheduler, especially for low to moderate queue rates, and is even able to match PF-based exhaustive search fairness.

Another advantage of multistream scheduling is that users can be scheduled more often than with single stream algorithms. Throughputs are therefore smoother, less bursty, which can be an advantage for multimedia services like audio and video streaming.

Similar results have been accepted for oral presentation at the 15th Annual Symposium of the IEEE/CVT Benelux Chapter in Antwerpen on November 2008 [25].

Chapter 4

Multiple antenna heuristics

Multiple antenna transmissions are nowadays well investigated. We can distinguish Single Input Multiple Output (SIMO), with multiple receive antennas, Multiple Input Single Output (MISO), with multiple transmit antennas, and MIMO, with multiple antennas at both transmit and receive sides. Depending on their use, multiple antennas can improve reliability, throughputs, or limit interferences. The idea we would apply here is to extend our code orthogonality factor to other dimensions such as antenna patterns.

There are many ways to use multiple antennas. The next section presents them, before focusing on a particular one.

4.1 Different multiple antennas usages

We are presenting here three major multiple antenna usages, namely Antenna Diversity, Multiplexing and Beamforming.

4.1.1 Antenna Diversity

A first MIMO technique is *Spatial* or *Antenna Diversity* [26, p. 78], which combats fading, like for example Space-Time Block Codes (STBC) [27]. When using N_{Tx} transmit antennas and N_{Rx} receive ones, there are actually $N_{Tx} \cdot N_{Rx}$ different paths. This number of paths is called the diversity degree.

If we transmit the same information on all these paths, we can assume the information will be easier to decode than if it would only be transmitted on a single faded path, since multiple paths are impaired with different fadings. Indeed, [26] explains that at high SNIR, the error probability of a SISO transmission depends on $\frac{1}{SNIR}$, whereas this error probability depends on $\frac{1}{SNIR^{(N_{Tx} \cdot N_{Rx})}}$ for MIMO transmissions.

As far as scheduling is concerned, antenna diversity does not bring any new challenge. The same information is transmitted through all multiple paths, so allocations can be performed just like in SISO scenarios. This is just the channel reliability which is improved through diversity.

4.1.2 Multiplexing

Instead of limiting fading, we can take advantage of it, with *Multiplexing* techniques [28] like V-BLAST [29]. If the path of each transmit-receive antenna pair is independent from each other, we obtain some parallel transmission channels. And each of these channels is able to transmit different data simultaneously. When the MIMO channel matrix is well conditioned, this multiplexing gain can reach $\min\{N_{Tx},\ N_{Rx}\}$.

Multiplexing on its own therefore improves capacity, but does not turn scheduling more complex. For each user, multiplexing makes several logical parallel channels, according to its channel matrix, but these logical channels can be allocated like a single high capacity SISO channel. Inter-user interference is not increased by multiplexing.

4.1.3 Beamforming

Another multiple transmit antennas technique is *Beamforming* [30, pp. 285-291], also known as smart antennas. By carefully selecting powers and phase shifts of different antennas transmitting the same signal, we can generate destructive interferences (signals of opposite phase) into some directions, whereas constructive interferences (signals with same phase) are generated towards other directions. So power can be focused in beams of particular directions while limiting interferences towards other directions. Beam orientation depends on phase shifts, while beam shape depends on the number and the disposition of antennas.

Beamforming allows to transmit into N_{Tx} different directions independently. So users in these directions can use the same codes, on the same frequency bands simultaneously without interfering with each other.

There are two beamforming modes [21, p. 344]. We can either use pre-computed weights, corresponding to pre-determined transmit directions, and transmit towards the closest direction to the user (*fixed beamforming*), or we can compute the weights corresponding to the effective user's direction (*specific beamforming*).

Multistream scheduling can therefore be performed based on users' position. This is spatial diversity. Codes and frequencies can be spatially shared. This imply some restrictions on simultaneous users' positions,

which can be modeled by some spatial orthogonality factor.

One can also imagine sharing each of the simultaneous beams by HSDPA codes. So users could be gathered into groups of same transmit direction, to share the same beam. Groups should be scheduled first, to steer beams, then users of each beam should be scheduled just as with SISO transmissions.

4.1.4 Choice of multiple antennas technique

Selecting diversity, multiplexing or beamforming techniques depends on channel quality, and on the knowledge we have about it.

If the channel is spatially decorrelated, and the transmissions of the multiple antennas take different paths, one can use diversity or multiplexing techniques. If, on the other side, the channel is spatially correlated, one can better select beamforming. Large antenna spacing is required for diversity and multiplexing techniques, while beamforming requires small antenna spacing.

If the transmitter does not have information about the users' location, it will not be able to steer a beam towards him, so beamforming can not be performed. Multiplexing also require channel knowledge to encode the simultaneous independent flows. On the other hand, diversity techniques do not require any Channel State Information (CSI) on the transmitter side.

Different techniques can also be combined. For example, [31] presents a strategy mixing multiplexing and beamforming through a wide array of narrow arrays. Combining antenna diversity and beamforming has also been studied by [32, 33, 34]. Multiplexing and diversity can also be jointly used [35], to bring a trade-off between reliability and increased throughput.

4.2 A beamforming channel model

This section presents a beamforming channel model, and corresponding interference mitigation. We do not detail here how the beamforming technique works, neither how to compute phase shifts to steer beams. This would lead us far away from our initial scheduling problem. In this work, we are more focused on clever resource allocations and beamforming effects than by beamforming implementation.

Whereas the SISO channel of a user U_k can be handled by a single

scalar value H_k, $(N_{Tx} \times N_{Rx})$ MIMO channels are handled by matrices:

$$H_k = \begin{pmatrix} H_k^{(1,1)} & \cdots & H_k^{(1,N_{Rx})} \\ \vdots & \ddots & \vdots \\ H_k^{(N_{Tx},1)} & \cdots & H_k^{(N_{Tx},N_{Rx})} \end{pmatrix} \ (N_{Tx} \times N_{Rx}) \tag{4.1}$$

MIMO channel capacity C_k can also be estimated through Shannon's theorem [14], such that

$$C_k = BW \cdot \log_2 \det \left(I_{N_{Rx}} \ + \ SNIR_k \cdot \frac{H_k \cdot H_k^H}{N_{Tx}} \right) \tag{4.2}$$

where $I_{N_{Rx}}$ is a $(N_{Rx} \times N_{Rx})$ identity matrix, and $(\cdot)^H$ denotes the complex transpose matrix operator.

The signal power S_k is given by

$$S_k = \mathrm{tr}\left(H_k \cdot H_k^H\right) \ \frac{P_k}{d_k^\gamma} \tag{4.3}$$

where P_k is still the power of the user U_k, d_k^γ is the pathloss, and $\mathrm{tr}(\cdot)$ denotes the trace matrix operator.

As in SISO, there are some ISI caused by multipath fading. They still can be handled by the β parameter we defined through Eq. (2.5) in Section 2.2:

$$ISI_k = \beta_k \cdot \mathrm{tr}\left(H_k \cdot H_k^H\right) \ \frac{P_k}{d_k^\gamma} \tag{4.4}$$

There are also intracell MAI, due to overlapping transmissions. The MAI perceived by a user U_k can be estimated by

$$MAI_k = \sum_{i \neq k} G_{i\,k} \cdot \mathrm{tr}\left(H_k \cdot H_k^H\right) \ \frac{P_i}{d_k^\gamma} \tag{4.5}$$

where $G_{i\,k}$ is the fraction of the transmit power of the i^{th} beam which is emitted towards user U_k. This is actually a spatial orthogonality factor. The authors of [36] present a simplified model for these factors:

$$G_{i\,k} = \begin{cases} 1 - \frac{|\theta_i - \theta_k|}{2\ \Theta_B(\theta_k)} & \text{if } \theta_k \in \left[\theta_i - \Theta_B(\theta_k)\,,\ \theta_i + \Theta_B(\theta_k)\right] \\ G_0 & \text{if } \theta_k \notin \left[\theta_i - \Theta_B(\theta_k)\,,\ \theta_i + \Theta_B(\theta_k)\right] \end{cases} \tag{4.6}$$

where θ_k is the direction from the base station to the user U_k, $2 \cdot \Theta_B(\theta_k)$ is the half-power beamwidth of the beam pointing towards θ_k. Outside the beam, some residual power G_0 is still transmitted, such as beam orthogonality is actually a near orthogonality relation.

Actually, Θ_B depends on the transmit direction. The closer to the boresight of the antenna it transmits, and the narrower the beam will be. This can be approximated by

$$\Theta_B(\theta_k) \approx \frac{\Theta_B}{\cos(\theta_k)} \tag{4.7}$$

As [36] mentions, for four transmit antennas, we have

$$\begin{array}{c} \Theta_B = 15^\circ \\ G_0 = -12dB \end{array} \tag{4.8}$$

As in SISO, the MAI_{inter} can also be estimated through G-factor.

$$MAI_{inter} = \frac{ISI + MAI_{intra}}{G_{factor}} \tag{4.9}$$

The SNIR can therefore be obtained by:

$$SNIR_k = \frac{S_k}{\sigma + ISI_k + MAI_k + MAI_{inter}} \tag{4.10}$$

$$= \frac{S_k}{\sigma + (ISI_k + MAI_k)\left(1 + \frac{1}{G_{factor}}\right)} \tag{4.11}$$

$$= \frac{P_k}{\frac{\sigma \cdot d_k^{\gamma}}{\mathrm{tr}\left(H_k H_k^H\right)} + \left(\beta_k P_k + \sum_{i \neq k} G_{ik} P_i\right)\left(1 + \frac{1}{G_{factor}}\right)} \tag{4.12}$$

Knowing the SNIR, the channel capacity can be derived from Eq. (4.2), and finally the channel capacity is truncated to get the throughput, as in SISO scenarios:

$$B_k = \min\left\{C_k\,,\ \frac{Q_k}{TTI}\right\} \tag{4.13}$$

4.3 A beamforming heuristic

There are several beamforming strategies. Beams can be randomly steered, and then be opportunistically allocated. This is the Random or Opportunistic Beamforming technique, introduced in [37], which [38, 39], among others, apply to the OFDMA framework. Instead, beams can be steered with some Zero-Forcing (ZF) or Minimum Mean Square Error (MMSE) techniques into greedily selected users' direction, exploiting spatial orthogonality [40, 41, 42]. Beams can also be steered according to a pre-defined codebook, where users only feed back the beam index that suits them the most. As far as the codebook design is concerned,

the authors of [43] proposed a Grassmannian line packing solution to the problem.

This section discusses some heuristic algorithms to allocate beams in multi-user MIMO (MU-MIMO) transmissions context, considering only one receive antenna per user equipment. As result, each beam only consists of a MISO transmission.

Scheduling beams is a bit different than scheduling codes. Indeed, there are at most N_{Tx} beams, and users can not receive more than one beam, as several beams transmitting in the same direction would cause a high level of interference. Moreover, we have to pay attention to space orthogonality. Interference depends on which particular users are transmitting. Users selection should be done carefully, but as our computations of Appendix C on the probability to find users with non overlapping beams demonstrate, even among a limited amount of users, it is possible to find some of them which would not cause too much interference to one another.

Let us point out that we do not consider multiplexing beams through orthogonal HSDPA codes with this beamforming heuristic. All codes are reused from one beam to another, and each scheduled user benefits from the 15 HSDPA codes of its beam.

4.3.1 Heuristic

As in SISO, the users' selection is a multi objective problem, where we have to maximize contradictory objectives of throughput and fairness. The scheduler we are going to design therefore tries to provide a good trade-off between those objectives. For an optimal resource allocation, we would have to evaluate a given criteria, such as PF, for all possible groups of users, for every resource repartition. This can not be done at TTI scale. So as in SISO, we aim at designing a PF based heuristic, taking advantage of multiuser diversity to allow good throughputs while providing some fairness.

We thought about two separate strategies for the beam allocation. The first one is to sort users according to their PF criteria, without taking into account interferences. Then we can compare the allocation of the full transmit power over a single beam steered towards the best user (specific Beamforming strategy) vs two beams towards the best two users, vs three beams towards the best three users, and so on, until N_{Tx} beams. The issue with this strategy is that users are sorted out without taking into account any MAI. Two users can have very good quality channels, but if they are close to each other, beams will overlap, and a lot of interference will ruin transmissions.

A different strategy would be to greedily loop on resources, beam after beam, trying to find the best user for each of them. Beams would be steered towards their user, in a specific Beamforming strategy. A first step would be to find the user which would benefit the most from all the power focused on a single beam. Then we should find another user, which would benefit the most from half of the transmit power focused on another beam, taking into account MAI_{intra}. And so on, we would allocate beam after beam, paying attention to MAI_{intra} to preserve already steered beams from too much interference. Afterwards, we would have to compare which amount of beams maximizes the sum of scheduled users PF criteria.

This second strategy has the advantage to consider MAI_{intra} when selecting users. It avoids to select users who are too close from each other. When selecting a new user, it takes into account already selected ones, trying to limit interference the new user would undergo from already selected ones. But interferences due to overlapping beams are symmetric. If new users are not interfered too much by already selected ones, they are neither degrading too much the transmissions of already selected users. This beam allocation strategy is synthesized by Algorithm 4.

Algorithm 4 Multistream Beamforming HSDPA PF algorithm

for beam $b = 1 : N_{Tx}$ **do**

Find user U_{k_b} such as

$$k_b = \arg\max \left\{ \frac{\min\left\{C_k\left(\frac{P_{HSDPA}}{b}\right),\ Q_k\,/\,TTI\right\}}{\overline{B_k}} \right\}$$

where

$$C_k\left(\frac{P_{HSDPA}}{b}\right) = BW \cdot \log_2\left(I_{N_{Rx}} + SNIR_k\left(\frac{P_{HSDPA}}{b}\right)\frac{H_k \cdot H_k^H}{N_{Tx}}\right)$$

with $SNIR_k\left(\frac{P_{HSDPA}}{b}\right)$ derived from Eq. (4.12), considering a transmit power of $\frac{P_{HSDPA}}{b}$ per beam.

Evaluate PF criteria of the b beams solution

$$PF_b = \sum_{u=1}^{b} \frac{\min\left\{C_{k_u}\left(\frac{P_{HSDPA}}{b}\right),\ Q_{k_u}\,/\,TTI\right\}}{\overline{B_{k_u}}}$$

end for

Schedule b^* beams such as $b^* = \arg\max\left\{PF_b\right\}$

Parameter	Value
Number of users	8 users
Traffic model	Fixed packet arrival rate, packets of uniformly distributed size
Channel model	SCME
Bandwidth	5MHz
Carrier Frequency	2GHz
Resource to share	1 to 4 beams
User location	Between 35 and 500m away from access point
User velocity	30kmph
Transmit power	32W
Noise power	-101.2 dBm
Interference model	ISI + Overlapping beams + Intercell interference estimated through G-factor
Antenna pattern	4×1 MU-MISO
TTI duration	2ms
Run duration	100 TTIs
Number of runs	400 runs

Table 4.1: Parameters for CDMA MIMO computations.

As far as knowledge requirement is concerned, we have to assume base stations know channel matrices, pathloss, angular position and even queue rates of each user. This represent a significative feedback load which should be reduced to allow practical implementation. Moreover, sharing beams to multiple users through HSDPA codes would also be useful. Nevertheless, this illustrates the beamforming gains one can obtain from multiple antenna transmission and code reuse between orthogonal beams.

4.3.2 Performance evaluation

We implemented Algorithm 4 into MATLAB scripts to evaluate its performance. We compare the allocation of 1, 2, 3 or 4 simultaneous beams. We still use the SCME channel model [22], and consider 8 users moving at the velocity of 30kmph, spread out within one sector of a three-sector cell, uniformly distributed in a third of a ring, between 35 and 500m away from the base station. Probability density functions about users' position can be found in Appendix B. We model 4×1 MISO transmissions, with 5MHz bandwidth, 32W of HSDPA transmission and -101.2dBm

of noise, as [21, Table 12.7] mentions. As in SISO, to model the queue rates, we assumed a fixed arrival packet rate of 1 packet per TTI, and a uniformly distributed size of packet. We considered the five mean queue rates presented in Table 3.1. All these parameters are summarized within Table 4.1.

Throughput results

To evaluate the throughput gain of multistream scheduling in beamforming scenarios, we compared our algorithm with a variation of it only considering a single beam transmitting with full power. We also considered as benchmark on our heuristic an exhaustive search trying to maximize the PF criteria of the allocation of 1 to 4 simultaneous beams.

The average sector throughputs are presented in Table 4.2, while Table 4.3 shows throughput gains of multistream allocation on a single beam allocation. Cumulative probability density function of users' throughputs for the different queue rates are presented at Fig. 4.1.

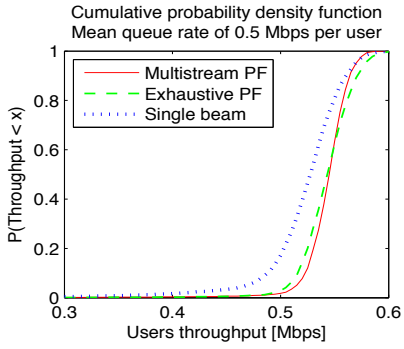
Users mean queue rate	Multistream heuristic	Multistream exh. search	Single beam
0.5 Mbps	4.30 Mbps	4.30 Mbps	4.14 Mbps
1 Mbps	8.36 Mbps	8.35 Mbps	7.35 Mbps
2 Mbps	14.39 Mbps	14.02 Mbps	9.59 Mbps
4 Mbps	19.04 Mbps	17.37 Mbps	10.54 Mbps
8 Mbps	20.60 Mbps	18.21 Mbps	10.65 Mbps

Table 4.2: CDMA MISO sector throughputs.

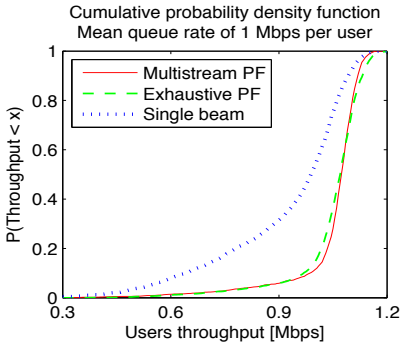
Users mean queue rate	Heuristic gain	Exh. search gain
0.5 Mbps	1.04	1.04
1 Mbps	1.14	1.14
2 Mbps	1.50	1.46
4 Mbps	1.81	1.65
8 Mbps	1.93	1.71

Table 4.3: CDMA MISO multistream gains on throughputs.

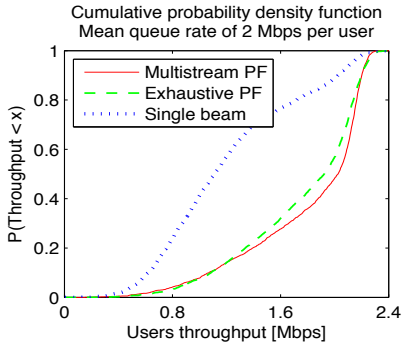
As one can see, for low queue rates, the multistream scheduling gain is not important. Actually, with low queue rates, channel capacities are enough to handle users queues, even with a single beam. But when queue



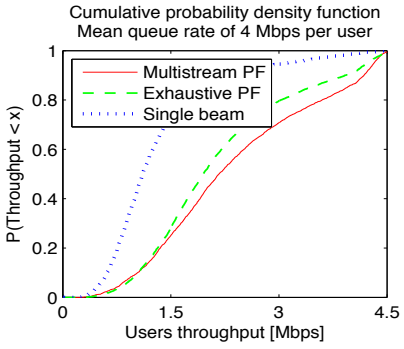
(a) 0.5 Mbps



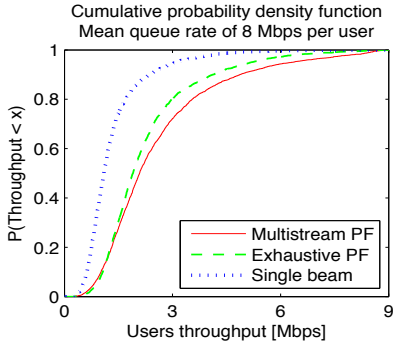
(b) 1 Mbps



(c) 2 Mbps



(d) 4 Mbps



(e) 8 Mbps

Figure 4.1: Cumulative probability density function of the users throughput, for 8 users with a mean queue rate of respectively 0.5, 1, 2, 4 and 8 Mbps.

rates become larger, the capacity of a single beam is overwhelmed, and multistream scheduling provides some throughput gain. Actually, this gain can be approximated by

$$BF_{gain} = \frac{N_{Tx}}{\log_2(N_{Tx})} \tag{4.14}$$

Indeed, N_{Tx} transmit antennas may handle N_{Tx} beams, so capacity should be multiply by a factor N_{Tx}. But the power allocated to each beam is divided by N_{Tx}, therefore their capacity is reduced by a factor $\log_2(N_{Tx})$. And actually, for $N_{Tx} = 4$, $BF_{gain} = \frac{4}{\log_2(4)} = 2$. Of course, this is only an approximated gain, which does not consider the issue of overlapping beams. But if users' direction are different enough, non overlapping beams can be found, like in the simulations we have performed.

As far as our heuristic competes with the exhaustive search, they provide the same throughput at low queue rates. When considering larger queue rates, our heuristic actually outperforms the exhaustive search throughput. Indeed, the exhaustive search maximizes the total PF, providing more fairness, at the expense of the total throughput, while our heuristic selects the best user, allows it to benefit from all resource, and only allocates leftover resources to other users.

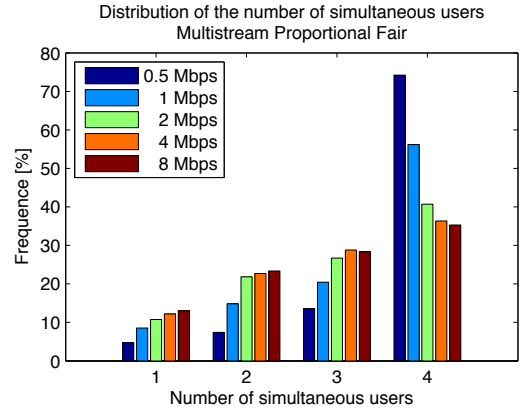
Number of simultaneous beams

We also evaluated the beam repartition of multistream schedulers. Averaged results are presented by Fig. 4.2 for different user mean queue rates. At low queue rates, our heuristic schedules a maximum amount of simultaneous users, but as queue rates are increased, it tends to schedule fewer simultaneous users. However, even with 8 Mbps queue rate per user, it still schedules 4 simultaneous users one TTI out of three in average.

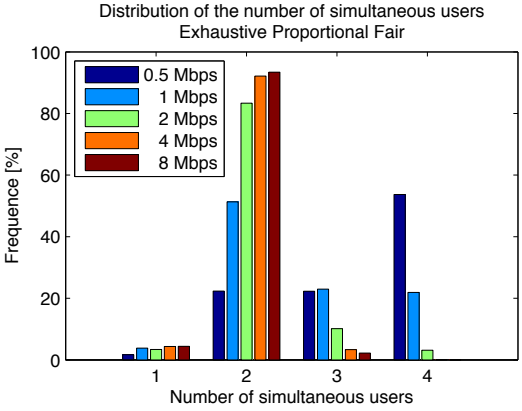
The exhaustive search also tends to schedule more simultaneous users when users queue rates are low, but as the queue rates are increased, most of the time it schedules two simultaneous users. The single beam solution is nearly never selected by the exhaustive scheduler.

Fairness results

We have also evaluated the fairness of the beamforming algorithms. Figure 4.3 compares the Jain Index of the multistream algorithms with the one obtained when only considering a single beam benefiting from the full transmit power.



(a) Heuristic



(b) Exhaustive search

Figure 4.2: Number of simultaneous beams, for 8 users with a mean queue rate of respectively 0.5, 1, 2, 4 and 8 Mbps.

Actually, when queue rates are as low as 0.5Mbps per user, the cell capacity is sufficient for all algorithms to fully serve all users, leading to the same perfect fairness. When queue rates are increased, multistream capacity manages to preserve more fairness. At high queue rates, the exhaustive algorithm, maximizing a total PF criteria provides more fairness than our algorithm, which, even if it uses PF to sort users, does not really try to maximize it. This better fairness for the exhaustive algorithm is obtained at the price of a lower throughput, as we already discussed.

Finally, as is SISO CDMA, we retrieve the general trend that the higher queue rates are, and the less fair algorithms become.

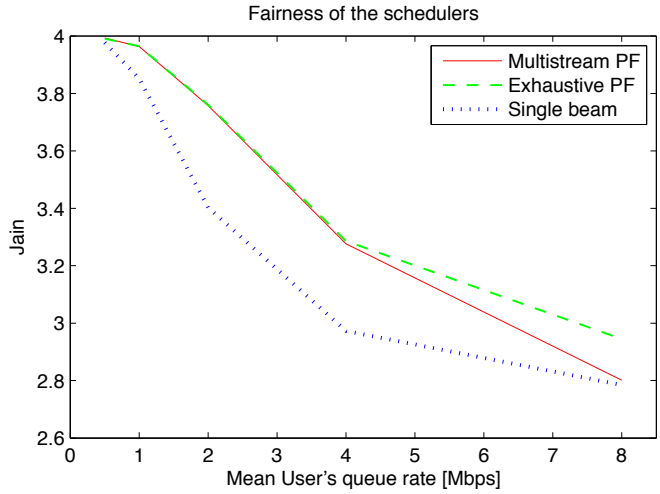


Figure 4.3: Fairness of the MIMO schedulers, depending on the users' mean queue rate.

4.4 A GoB heuristic

As we have just shown, MU-MIMO beamforming technique is able to provide better radio efficiency through resource reuse between spatially orthogonal users. Throughput and fairness gains are especially obtained with large user queue rates, while in SISO the largest gains were obtained at low queue rates. To evaluate the gain combining both approaches, this section considers beamforming strategies where the HSDPA codes of a beam can be split to several users.

Sharing a beam with multiple users through HSDPA codes introduces some new challenges. First, users of close transmit direction have to be gathered into groups. Then the scheduler has to select groups towards whom it will steer beams, and finally, within each selected group, codes have to be allocated to users. To present a simple solution to this problem, we considered the GoB technique [44, pp. 51-55] we discuss hereafter.

4.4.1 The Grid of Beams technique

As we already mentioned in Section 4.2, 4 transmit antennas can generate up to 4 beams whose half power beamwidth is 30° [36]. So the 120° sector of a 3-sector cell can be covered with a series of 4 beams. Actually, to improve performance, we considered 2 series of beams, with a second one shifted from a half beamwidth, as presents Fig. 4.4.

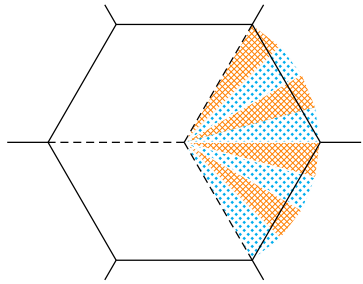


Figure 4.4: GoB covering of a 120° cell-sector with 4-Tx antennas

Figure 4.5 shows the beam pattern of the GoB, according to the beam model of Eq. (4.5). The two series of beams (orange or blue) can be time alternated. As one can see, whatever the position of the user is, between 75% and 100% of the emitted power is effectively transmitted towards him. And thanks to beam orthogonality, all 15 HSDPA codes can be reused within each beam, with inter-beam interferences limited to 6% of the other beams transmit power. Within each beam, resources can be shared exactly as we performed in SISO.

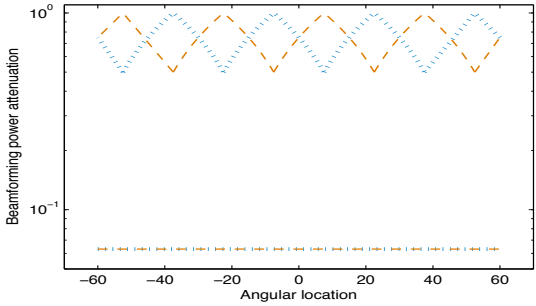


Figure 4.5: Beam pattern of our GoB covering of a 120° cell-sector with 4-Tx antennas

4.4.2 Heuristic

The first allocating step is to gather users into beams. One can imagine, after listening to transmission from every beam, that users feed their best beam index back to the base station, which can therefore gather them for beamforming. And then, on a second step, the scheduling algorithm only has to share the 15 HSDPA codes within each beam of the current series separately, which can be done according to a PF strategy, as in SISO. The two series of beams can be simply time-alternated in a RR kind of way.

This strategy is summarized by Algorithm 5.

Algorithm 5 HSDPA GoB PF algorithm

Considering a total transmit power P_{HSDPA} per beam,

for all users U_k **do**

Identify its best beam index i;

Evaluate SNIR:

$$SNIR_k = \frac{G_{i\,k} \cdot \mathrm{tr}\left(H_k \cdot H_k^H\right) \cdot \frac{P_{HSDPA}/15}{d_k^\gamma}}{\sigma + (\beta_k + 3G_0) \cdot \mathrm{tr}\left(H_k \cdot H_k^H\right) \cdot \frac{P_{HSDPA}}{d_k^\gamma} \cdot \left(1 + \frac{1}{G_{factor}}\right)}$$

Evaluate potential throughput of one code:

$$B_k = \min\left\{BW \cdot \log_2 \det\left(I_{N_{Rx}} + SNIR_k \cdot \frac{H_k \cdot H_k^H}{N_{Tx}}\right),\ \frac{Q_k}{TTI}\right\}$$

end for

for each beam of the current series **do**

while there are still unused codes and users to be scheduled **do**

for all users U_k of the beam **do**

Find the desired number of codes NC_k for each user yet to schedule, with NC_{unused} standing for the number of yet unallocated codes:

$$NC_k = \underset{N \le NC_{unused}}{\arg\max} \left\{N \cdot B_k \ \text{ such as } N \cdot B_k \le Q_k\right\}$$

Evaluate the PF criteria of each allocation: $PF_k = \frac{NC_k \cdot B_k}{\overline{B_k}}$

end for

Allocate to the user k^* the NC_{k^*} he desires, where

$$NC_{k^*} = \min\left\{\arg\max_k\{PF_k\}\ ,\ NC_{unused}\right\}$$

end while

end for

4.4.3 Performance evaluation

This algorithm has been implemented into MATLAB scripts, and we compare its throughput and fairness performances against our beamforming algorithm, where the 15 codes of a beam are allocated to the same user, and against our SISO algorithm, where codes are allocated one by one to users, but are not reused through any beamforming strategy.

Since 8 beams are considered with our GoB strategy, modeling 8 users as previously seemed to us not enough. We considered instead 40 users, with queue rates five times lower as in our previous SISO and MU-MIMO beamforming computer simulations. Table 4.4 details all our simulation parameters.

Parameter	Value
Number of users	40 users
Traffic model	Fixed packet arrival rate, packets of uniformly distributed size
Channel model	SCME
Bandwidth	5MHz
Carrier Frequency	2GHz
Resource to share	4 simultaneous beams split through 15 HSDPA codes
User location	Between 35 and 500m away from access point
User velocity	30kmph
Transmit power	$\frac{32}{4}$W per beam
Noise power	-101.2 dBm
Interference model	ISI + Overlapping beams + Intercell interference estimated through G-factor
Antenna pattern	4×4 MU-MIMO
TTI duration	2ms
Run duration	100 TTIs
Number of runs	100 runs

Table 4.4: Parameters for GoB CDMA computations.

Throughput results

Cumulative probability density functions of throughputs reached with each algorithm are presented in Fig. 4.6, with GoB denoting the algo-

rithm sharing beams through HSDPA codes, SISO where there are only the 15 HSDPA codes to share, and BF where there are only 1 to 4 simultaneous beams, which can not be shared between simultaneous users.

At the really low queue rate of 0.1 Mbps per user, our SISO and GoB strategies provide the same throughputs, larger than what the beamforming algorithm allows, as Fig. 4.6(a) shows. Indeed, the beamforming strategy can only schedule up to 4 simultaneous users, while other algorithms, able to share HSDPA codes, can serve more simultaneous users. And since throughputs are limited by these low queue rates and not by channel capacity, serving more simultaneous users provides larger throughputs.

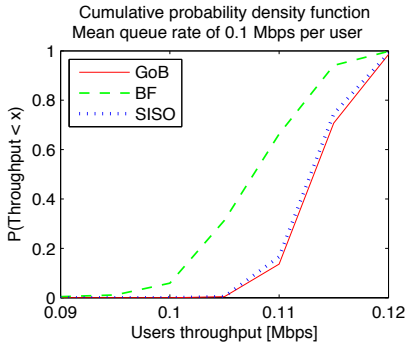
When considering results of Fig. 4.6(b), with 0.2 Mbps of mean user's queue rate, the MIMO strategies becomes better, such that BF manages to match SISO throughput, while GoB throughput becomes larger. This result seems coherent with our previous beamforming analysis, which showed that the gain of using multiple beams appears when queue rates are large enough. And as the comparison between GoB and BF is concerned, since GoB allows to share beams though HSDPA codes, and not BF, GoB allows larger throughputs than BF, which is coherent with our previous SISO analysis.

Then with the larger queue rates of Figs. 4.6(c), 4.6(d) and 4.6(e), SISO throughputs become poorer than those considering multiple simultaneous beams. Indeed, as we already mentioned, larger throughputs benefit to beamforming strategies. When it comes to comparing GoB and BF throughputs, they become pretty similar to each other, which is understandable since we have already shown that with the largest throughputs, all codes tend to be allocated to the same user.

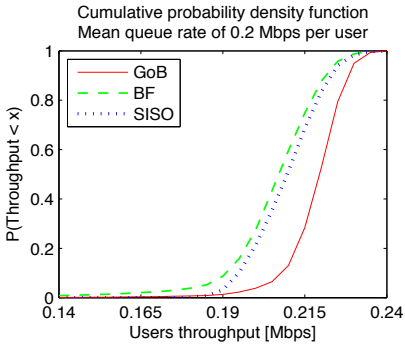
Fairness results

Jain index of the three algorithms for the different queue rates is presented by Fig. 4.7. As usual, fairness becomes lower as queue rates are increased. GoB and SISO, for low to moderate queue rates, are quite less fair than BF. Better fairness for BF vs SISO can be explained by larger BF capacity, as in our beamforming analysis.

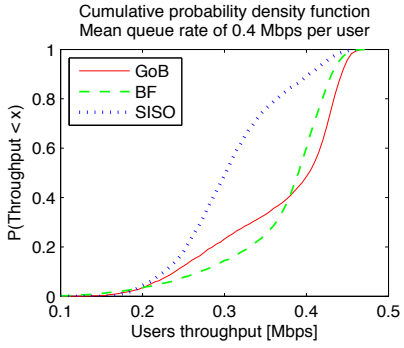
As far as the lower fairness of GoB, this probably comes from the grid design. Depending on its location on the grid, a user only gets 75% to 100% of the transmit power, and though our computations consider a Doppler effect caused by user velocity, user positions are actually fixed. Users close to the boresight of their beam are therefore favored.



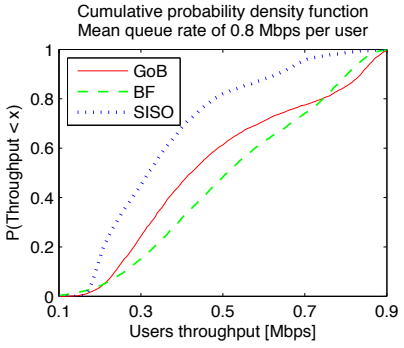
(a) 0.1 Mbps



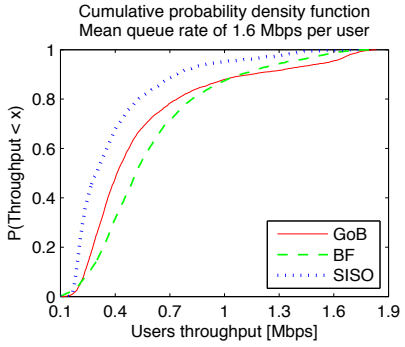
(b) 0.2 Mbps



(c) 0.4 Mbps



(d) 0.8 Mbps



(e) 1.6 Mbps

Figure 4.6: Cumulative probability density function of the users throughput, for 40 users with a mean queue rate of respectively 0.1, 0.2, 0.4, 0.8 and 1.6 Mbps.

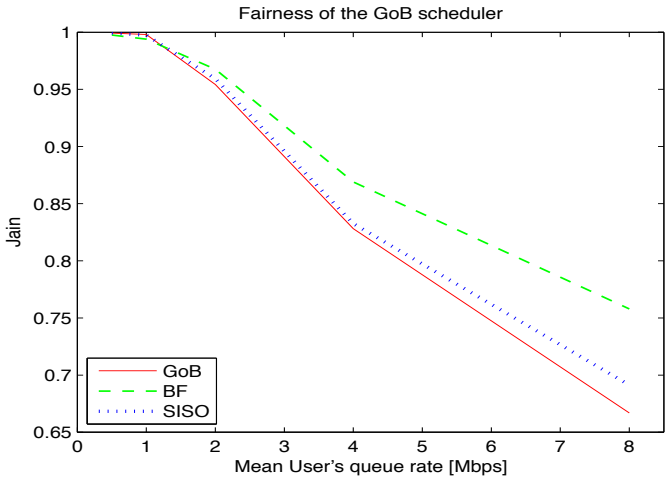


Figure 4.7: Fairness of the schedulers, depending on the users' mean queue rate.

4.5 MU-MIMO concluding remarks

This chapter presented multistream scheduling algorithms in MU-MIMO HSDPA environment. A first beamforming algorithm we designed is based on a PF criteria for its trade-off between user fairness and cell throughput. It manages to handle MAI_{intra} through spatial orthogonality factor, and shares transmit power to 1 to N_{Tx} users through unshared simultaneous beams. Compared to a single beam scheduler, it is able to improve cell capacity through HSDPA code reuse within each beam, especially at high queue rates, as well as fairness. A bound on capacity gain has been set up, and seems reachable.

We then design another MU-MIMO scheduler, based on a GoB strategy. This enables us to cover a sector cell with multiple fixed beams, which can be shared to multiple users through the HSDPA codes, with all codes reused within each beam. Combining code and beam multiplexing techniques allows us, as we have shown, to benefit from both gains. At low queue rates, GoB benefits from the code multiplexing gains we already obtained in SISO, while at high queue rates, it benefits from code reuse within each beam. As far as fairness is concerned, some issues, depending on user position according to the grid design, arise.

Results similar to those of the beamforming analysis of Section 4.3 have been presented at the 6th Management Committee Meeting of COST2100 at Lille (France), in October 2008 [45].

Chapter 5

CDMA conclusions and perspectives

This chapter concludes our CDMA investigations. CDMA is the multiple access technique used in current 3G cellular networks, like the original UMTS (Rel'99) and its HSDPA/HSUPA evolutions. We presented the HSDPA standard and discussed about scheduling and interference mitigation in CDMA-based environment. We introduced a SISO heuristic scheduler, handling interference through orthogonality factor. Actually, we showed that in DL scenarios, interference only depends on the total transmit power, and not on who are the specific users to which the simultaneous transmissions belong. Our MATLAB simulations showed both throughput and fairness improvements compared to single stream schedulers, especially for low to moderate queue rates. Actually, with the largest queue rates, a single user is able to benefit from all resources, and there is no multistream gain.

We also considered multiple antenna heuristics. Among different multiple antenna transmission techniques, we focused on beamforming, which is the most challenging from a scheduling point of view. Indeed, since beamforming aims at focusing transmissions into limited directions, these simultaneous transmission directions have to be selected cleverly. With non overlapping beams, HSDPA orthogonal codes can all be reused within each beam, increasing the total cell capacity. The scheduling algorithm had to be adapted, since inter-beam interference depends on the position of each transmitting user, unlike MAI_{intra} in SISO environment. We managed to handle these interference through a spatial orthogonality factor. MATLAB simulations focusing on multistream through orthogonal beams only have been performed, and have shown multistream throughput gain, especially with the largest queue rates, while preserving the fairness of a PF allocation strategy.

Our orthogonality factors are summarized by Table 5.1. In SISO, MAI_{intra} is actually caused by desynchronization between flows, due to multipath echoes of transmissions. Both own transmission echoes and other users' transmission echoes cause interference modeled by the β factor we detailed in Eq. (2.5). When considering specific beamforming transmissions, where users do not share their beam, MAI_{intra} depends on users' position, and on overlapping beams. Orthogonality is therefore handled by the factors $G_{i\,k}$ we introduced in Eq. (4.6).

Table 5.1: CDMA orthogonality factors

Environment	Orthogonality factor
SISO	β (Eq. 2.5)
Beamforming	$G_{i\,k}$ (Eq. 4.6)

We also designed a GoB-based scheduling algorithm, considering a fixed beam covering of the sector cell, which enables us to combine code reuse between beams, and beams shared to multiple users through HSDPA codes. MATLAB simulations showed throughput gains, especially on a simple beamforming strategy at low queue rates, or especially on SISO strategy at large queue rates. The drawback is that fixed users' fairness is quite impaired by the grid design, which favor users closer to the center of their beam.

But as far as 4G standards are concerned, CDMA-based transmissions are phasing out. The multiple access technique of the two new standards, which are Mobile WiMAX and LTE, is no longer based on orthogonal codes, but on orthogonal frequency bands (OFDMA).

As [11] details, among other advantages, OFDMA allows handling large bandwidth, up to 20MHz, while UMTS CDMA can not handle more than 5MHz bandwidth. It also allows frequency domain scheduling, to schedule users on the frequency bands where their channel is the best, and therefore provides higher spectral efficiency. Moreover, interference management is much simplified, since through cyclic prefix, ISI can be removed. Multiple antenna transmissions, which are able to provide better throughputs, are also simpler to implement in OFDMA than in CDMA.

Furthermore, neither WiMAX nor its LTE counterpart mandate any scheduling algorithm. It is actually a degree of freedom for the equipment manufacturers. As a result, we then refocused our investigations on resource allocation in OFDMA-based WiMAX and LTE frameworks, which the next part of this thesis is about.

Part III

OFDMA: WiMAX & LTE

Chapter 6

WiMAX & LTE environments

Looking for better throughputs at high velocities, Institute of Electrical and Electronics Engineers (IEEE) and 3GPP both put forth their own 4G cellular network architecture. IEEE promotes a mobile version of its WiMAX standard, whereas 3GPP has standardized LTE in late 2008, and is now working on its LTE-Advanced evolution.

6.1 Radio resources

Both standards propose the two main duplexing techniques, namely TDD and FDD, and both consider OFDMA as multiple access technique. Simultaneously transmitting users are therefore not separated through orthogonal codes as in 3G networks, but through orthogonal frequency bands. Multiple access is achieved when users transmit over non overlapping frequency bands.

The WiMAX granularity allocation is the *subchannel*, which is composed of 24 frequency bands of 11kHz for a few symbols duration. Each symbol consists of 192 subcarriers transporting each a modulated symbol. Control information about the resource allocation map of a scheduling period, called frame, is broadcasted to users as both maps (DL-MAP and UL-MAP). In the same way, the resource unit of LTE is called a Physical Resource Block (PRB). In TDD systems, it corresponds to 12 (or 14) subcarriers of 15kHz for a 0.5 ms slot duration. The ratio between DL and UL can vary from 2:3 to 9:1 [46]. Table 6.1 summarizes this information.

Figure 6.1 compares resource allocation in WiMAX and LTE. The resource to allocate can be mapped as a rectangle whose dimensions are bandwidth and time. The surface to allocate is shrunk because of control information transmission (in blue on the figure), and because some fre-

Table 6.1: Radio Resource to allocate in Mobile WiMAX and LTE.

	Mobile WiMAX 10	LTE
Resource unit	Subchannel 24 x 10.9375 kHz during a frame	PRB 12 x 15 kHz during a TTI
Scheduling period	Frame = 5ms (48 symbols)	TTI = 1ms (12-14 symbols)
Resource signalling	DL-MAP UL-MAP	PDCCH PUCCH

quency bands should be avoided to limit intercell interference (in red). This is actually a simplified model. Chapter 9 specifically focuses on map allocation for WiMAX and presents a much more accurate resource model.

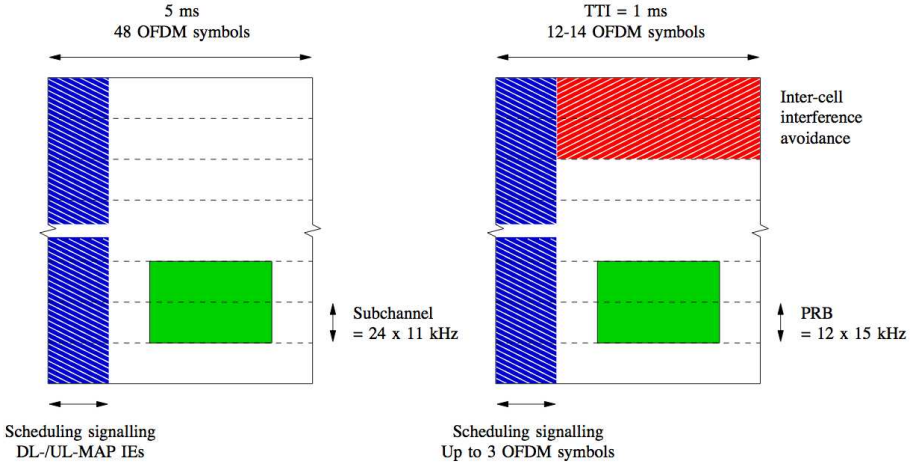


Figure 6.1: Radio Resource Allocation in Mobile WiMAX and LTE. The green block is an example of TDD resource allocation for a given session.

The subcarriers composing subchannels or PRBs are sufficiently narrow to allow using cyclic prefix before each OFDMA symbol, therefore protecting transmissions from ISI caused by multipath effect. However, when considering large bandwidth, each subchannel experiences its own channel response. So when it comes to frequency bands allocation, scheduling algorithms should allocate to users their best subchannels/PRBs, to take advantage from this frequency diversity. It however complicates the resource allocation, compared to CDMA orthogonal codes, which are all equivalent.

6.2 OFDMA scheduling

As far as scheduling and the allocation of these surfaces of resource are concerned, there is no mandated scheduling technique for WiMAX nor LTE. No information is provided about subchannels or PRBs allocation, which is actually a degree of freedom for the equipment manufacturers [47, 48]. OFDMA scheduling has already been studied, but most of the time considering simplifying assumptions, and neglecting channel feedback load, which can represent an important burden.

The work described in [49] models the WiMAX allocation resource problem and solves it through a heuristic. The authors of [50] present a few multistream extensions of the Max Weight scheduler. Another heuristic, proposed by the authors of [51], at first allocates to each user the subchannel that suits them the most, then the remaining subchannels are allocated to users with lower throughputs.

The allocation scheme described in [52] differentiates sessions with some QoS constraints from Best Effort (BE) ones. Sessions with QoS constraints are scheduled first, and receive subcarriers that both suit them the most and are the poorest for BE flows. Then BE sessions are scheduled with a PF-based strategy. Authors of [53] also presented a technique using different scheduling criterions for QoS vs non-QoS sessions.

A different traffic differentiation is presented by [2], where sessions are first divided between rushing vs non-rushing ones, according to their packets deadline. Rushing sessions are scheduled first, according to their deadlines, then remaining subcarriers are allocated trying to maximize total cell throughput.

Other scheduling algorithms are mentioned in [54], which underlines the difference between algorithms based on channel quality and algorithms based on sessions' queues. BE sessions can be scheduled according to a maximum utility strategy like maximum throughput, PF, etc. Sessions with QoS constraints are better scheduled considering packet deadlines (Earliest Deadline First (EDF)), or by combining channel quality and queue length or packet deadline (Log rule, Exponential rule).

There are also scheduling strategies based on a more geometrical approach. The Sequential Rectangle Placement (SRP) [55], which we will detail later on in Section 9.3.3, considers that each transmission occupies a set of subcarriers for a given time, and that sessions are sorted according to a predefined order. The authors of [56] presented a similar but more general technique, which orders sessions according to their QoS constraints. Even if this problem can be considered as particular instances of the Knapsack problem, authors of [55, 56] proposed quite efficient heuristics.

6.3 The feedback load issue

Transmissions through radio channel are impaired by fading, interference and noise. These impairments are captured by the CQI metric, which is fed back by the receiver to the transmitter. With wide channel bandwidth such as LTE's 20MHz, the channel is frequency selective. It means each subcarrier undergoes its own fading and interference, which is different from one user to another.

Most of the scheduling techniques try to benefit from this frequency diversity through allocating to users the frequency bands that suit them the most, in order to maximize cell throughput. However, knowing channel quality of each subchannel/PRB for each user for each frame/TTI implies a huge feedback traffic. If we consider a 5-bit feedback per subchannel/PRB like in [57], a LTE system requires 50kbps of feedback per user to handle 10 PRBs, and even 500kbps to handle 100 PRBs.

There are several strategies to reduce this huge feedback load. As [58] explains, feedback can be restricted for each user to its N best subchannels/PRBs. However, these best N subcarriers can be of poor quality, and not used by the scheduler, though they have cost feedback resources. We can also have two users reporting the same subcarriers as their N best ones, but may be some subcarriers not among the N best ones of the first user are better than the N best ones of another user. It would therefore be more efficient to schedule the first user on other subcarriers, about which the base station did not receive any feedback, to allow the other user transmit on its N best subchannels. To prevent these issues, [58] proposes a threshold strategy, where users only feed back CQI about subchannels/PRBs whose capacity is above a given threshold. The feedback load therefore depends on the selection of an appropriate threshold. Gathering adjacent subchannels/PRBs, which would roughly be of the same quality, has been proposed by [59]. The feedback load would therefore be reduced by the gathering factor. Some CQI compression techniques have also been proposed by [60].

Another strategy would be to consider averaged channels, handled by only a single averaged CQI value per user. Within WiMAX standard, there are the Partial Usage of Subchannels (PUSC) and Full Usage of Subchannels (FUSC) subcarrier gathering modes [61], where subcarriers are pseudo-randomly permuted before being gathered into subchannels. Since the subcarriers of each subchannel are distributed across the whole bandwidth, the channel response of PUSC/FUSC subchannels is somehow averaged out. Similarly in LTE, there is a feedback mode where users only report a single wideband CQI value [54]. Moreover, some OFDMA scheduling algorithms such as [55, 56] implicitly suppose aver-

aged channel. If each user benefits the same from all subchannels/PRBs, the surface of Fig. 6.1 can be shared regardless the location of each user allocation.

Further feedback load reduction schemes are discussed in [62], where authors consider CQI is reused during several scheduling periods, and discuss about the frequency of CQI updates. Since these strategies apply to any scheduler without favoring any of them, we are not going to consider CQI reuse to compare algorithms. So updated CQI values are used every scheduling period within our comparisons.

6.4 Traffic classes

We already mentioned QoS sessions and BE ones. Actually, WiMAX differentiates 5 kinds of session: UGS, rtPS, ertPS, nrtPS and BE [63]. The UGS and ertPS classes can be considered as real-time constant bit rate services, and should not be delayed by schedulers [64]. Their resource allocation, based on the maximum sustained traffic rate, should be decided at admission.

Similarly, LTE identify 9 traffic classes [65, Table 6.1.7]. Four of them are for Guaranteed Bit Rate (GBR) services, and five for non-GBR. Again, GBR flows should be scheduled in priority.

Chapter 7

A Treemap-based heuristic

In order to design a low feedback load requirement scheduling heuristic, we are going to only consider an averaged wideband feedback scheme. Indeed, it drastically reduces feedback load. Moreover, since for each user all subchannels have the same capacity, varying from one user to another, resource allocation can be done more accurately, therefore reducing resource wastes.

This however costs the loss of frequency diversity. Indeed, scheduling each user on its best frequency band leads to larger throughputs than scheduling on averaged channels. We are going to evaluate the cost of the loss of this frequency diversity gain, compared to the gain in feedback load reduction, and to the gain allowed by a simpler and more accurate resource allocation algorithm.

Considering each user benefits the same from all subchannels/PRBs, the resource to share can be seen as the surface of Fig. 6.1. Since scheduling then consists of dividing this surface, we thought about tiling and paving algorithms. One way to allocate surface to users can be to make tiles, whose size depends on user needs, and then to pave the surface with all these tiles. Actually, this tessellation is quite a heavy task [66]. Though simpler algorithms like [55, 56] have already been presented, we managed to design another algorithm, much more efficient, based on Treemap visualization.

Let us point out that each user can have different parallel sessions. One can easily imagine someone having a phone call while web browsing and downloading some files from the Internet. For simplicity sake however, we only consider here one session per user. Words *session* and *user* are therefore used indifferently.

7.1 Treemap visualization

Initially, treemap visualization has been designed in computer science to display hard disk drive usage, in order to ease the identification of large files. The tree structure of a file system is represented by rectangles fitted into each other, with size proportional to the disk usage of the directory they represent.

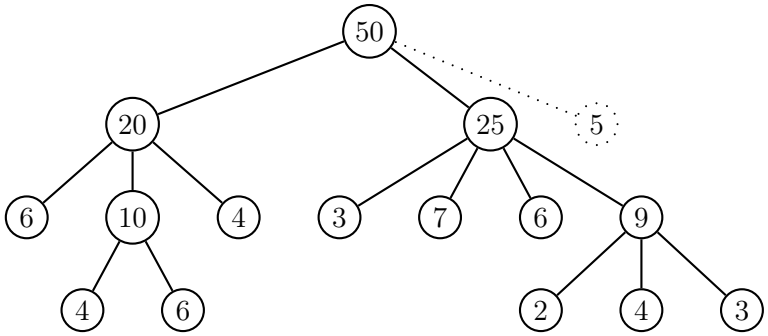
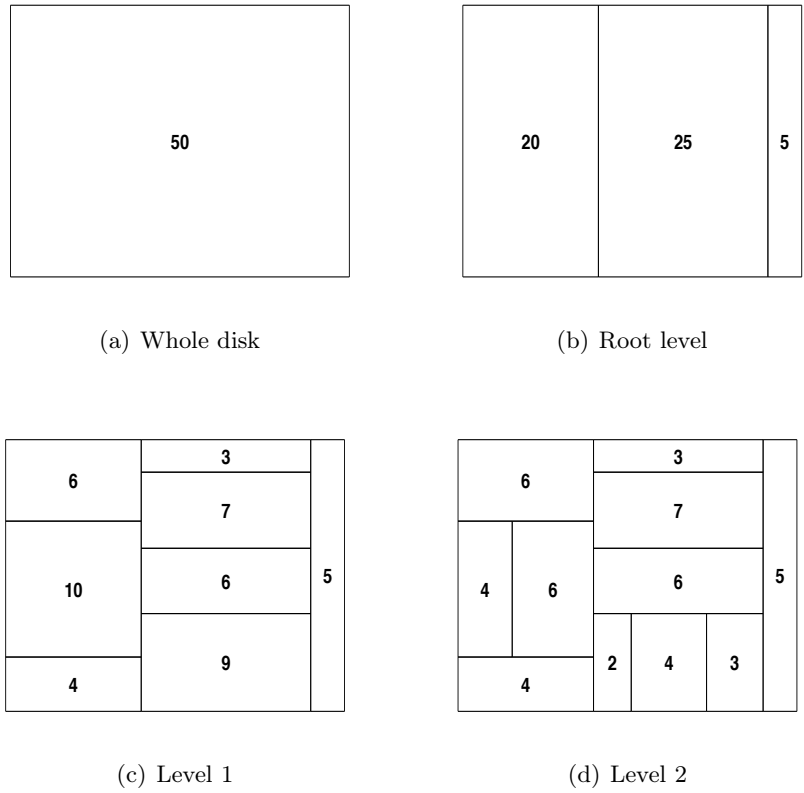


Figure 7.1: Example of a filesystem hierarchy.

A simple treemap visualization algorithm has been introduced in [67]. Figure 7.2 shows how it represents the filesystem of Fig. 7.1. Starting at root level, it divides the original rectangle of Fig. 7.2(a), representing the whole disk into different slices, with width proportional to the size of the first level directories. A special slice (of size 5 on Fig. 7.2(b)) is introduced to consider free space. And then, recursively, for each directory, the area is divided into smaller slices representing all sub-directories. Cuts should be done alternatively horizontally and vertically.

A drawback of this algorithm is that it tends to draw thin rectangles, whereas one would prefer nearly square rectangles (with aspect ratio close to one), for a better visualization. Indeed, it is easier to compare size of squared areas than of thin rectangles. The authors of [68] introduced another treemap algorithm, dividing each rectangle into smaller nearly square rectangles. The idea is to cut a first stripe into the rectangle to divide, and add rectangles into that stripe of growing width until aspect ratio can not be improved. Then another stripe is drawn, and so on. Stripes are cut alternatively horizontally and vertically.

For example, to represent the directory of size 9, composed of directories of size 2, 4 and 3, the squared treemap algorithm would produce the map of Fig. 7.3. Handling directories in descending order, it draws a



(a) Whole disk

(b) Root level

(c) Level 1

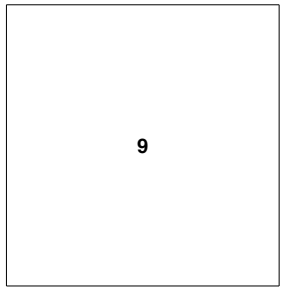
(d) Level 2

Figure 7.2: Example of a treemap visualization.

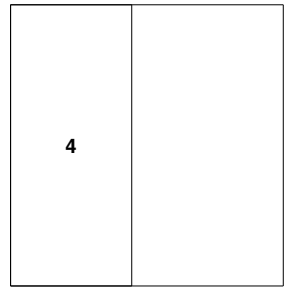
first stripe with directory of size 4 (Fig. 7.3(b)), then tries to add the directory of size 3 in the stripe (Fig. 7.3(c)). Since aspect ratio is increased, it tries to also add the directory of size 2 in the stripe (Fig. 7.3(d)). But here aspect ratio is decreased, so finally, the first stripe only contains directories of size 4 and 3, and another stripe is drawn for the directory of size 2 (Fig. 7.3(e)).

This squared treemap algorithm is a heuristic. An optimal result on aspect ratios can not be guaranteed, and counterexamples can be found. Nevertheless, its results are quite good, and even if it does not provide the ideal visualization, it runs in linear time. For better results, larger directories should be treated first.

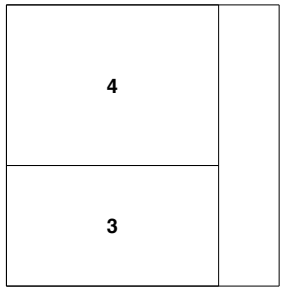
As far as scheduling through a treemap-based strategy is concerned, when a first scheduling step would have identified users granted for transmission, some treemap algorithm would be useful to actually map each selected user with a frequency-time area.



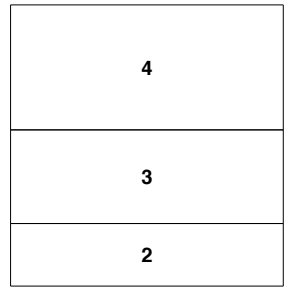
(a) Directory of size 9



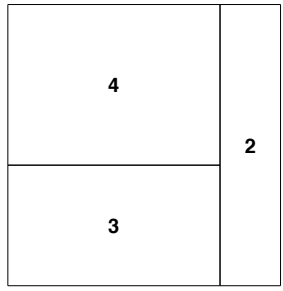
(b) Adding 4 to a first row



(c) Gathering 4 and 3 in the row, aspect ratio increased



(d) Gathering 4, 3 and 2 in the row, aspect ratio decreased



(e) A first row finally gather 4 and 3, and 2 is left in a second row

Figure 7.3: Example of square treemap repartition.

7.2 Additional constraints

The selection of transmitting users must comply with several constraints we list hereafter.

7.2.1 Traffic classes

As we already mentioned in Section 6.4, WiMAX differentiates 5 traffic classes, whereas LTE has 9 of them. For our heuristic model, we summarize all these classes into 3 different kinds of traffic, namely GBR, Real Time (RT) and BE. GBR and RT sessions have QoS guarantees, and should be scheduled first. GBR has constraints on throughput, whereas RT has no guaranteed throughput, but is delay sensitive.

7.2.2 Subscriber status

Network operators can also prioritize traffic based on users subscription profile, for example pre-paid vs post-paid users, or favor its own customers. In the following, we model it with an olympic-type of customer differentiation, with Gold, Silver and Bronze profiles. Within a traffic class, Gold packets should have higher priority than Silver ones, and Silver ones should have higher priority than Bronze ones.

7.2.3 Session selection

Treemaps are designed to map hard disk drive usage. They can only share resources. So before any treemap resource allocation can be done, we have to select users granted for transmission, ensuring their needs are compatible with the total channel capacity. This session selection should be done according to traffic classes and user status. GBR and RT sessions should be scheduled before BE ones, and for the same traffic class, Gold sessions should be scheduled before Silver and Bronze ones. Between sessions of the same class and user status, some scheduling criteria such as RR, PF, EDF, Max Throughput and so on, should be used to decide which session is going to be scheduled first.

7.2.4 Allocation granularity

Treemap algorithms draw rectangles continuously, but for OFDMA resource allocation, schedulers are constrained to discrete values, fitting the subchannel/PRB scale on frequency dimension, and fitting the symbol scale on time dimension. Therefore some rounding operations have to be considered.

7.2.5 Speed of convergence

On top of all these constraints, allocations have to be computed within the scheduling period duration. So while in WiMAX the algorithms have the 5ms of frame duration to provide the resource allocation of the next frame, in LTE they only have 1ms of TTI.

7.3 Algorithm

Due to heterogeneous traffic, this scheduling algorithm has multiple objectives. It has to ensure guaranteed throughput to GBR sessions, to minimize RT packet delays and to maximize BE throughput while considering subscriber status priority and providing fairness between BE sessions of the same category. The algorithm design should remain simple, and must only require a low feedback amount.

To bring an answer to this scheduling problem, we set up a 3-step algorithm. Considering a wideband averaged channel for each user, as well as its queue rate, our algorithm allocates radio resources providing QoS guarantees for GBR and RT sessions, while maximizing a PF criteria for BE trafic. A first step sorts sessions out according to a given hierarchy, while a second step assesses resource needs and selects users granted for transmission. A final third step allocates subchannels/PRBs to users with a treemap-based strategy.

7.3.1 Step #1 - Ordering sessions

A first scheduling step sorts the sessions, according to their traffic class, the time sensitivity of the packets, and the user status. GBR packets have to be scheduled first, with no delay. Cell admission control should ensure that guaranteed bit rate is achievable before accepting new GBR sessions, so the algorithm is able to serve them all. Then RT traffic should be scheduled. Because of the real-time nature of these packets, we selected an EDF strategy to sort them out. Finally, BE sessions are scheduled, starting with Gold ones, then Silver ones and Bronze ones. Between sessions of the same user status, we sorted them according to a PF rule. We opted for the PF criteria for its trade-off between user fairness and cell throughput.

So here is the three-level hierarchy used to sort sessions:

1) GBR
2) Non-GBR:
    1. Real-time (RT)
    2. Non real-time (BE):
        a) Gold
        b) Silver
        c) Bronze

Retransmissions could get higher priority, but we do not consider them in this work.

7.3.2 Step #2 - Assessing resource needs

The second step decides the number of scheduled sessions. Following the hierarchy obtained at Step #1, the algorithm evaluates for each session the amount of resource required to satisfy its throughput or delay constraints, until all resources are exhausted. These resources are then allocated at Step #3.

The quantum of resource allocation is the slot, which represents the usage of one subchannel/PRB for one symbol duration. Derived from its CQI, the slot capacity for each user has to be estimated. It actually depends on the SNIR, but also on the spectral efficiency of the transmission mode. We considered the work of [69], which lists Signal to Noise Ratio (SNR) thresholds to maintain a Bit Error Rate (BER) under 10^{-3} with QPSK, 16QAM and 64QAM modulations, using convolutional coding and Reed-Solomon block encoder [70]. These thresholds, obtained from Monte-Carlo simulations, are listed in Table 7.1.

Table 7.1: Set of MCS and required SNR thresholds for a BER $< 10^{-3}$

MCS	Spectral efficiency [bit/s/Hz]	SNR threshold [dB]
QPSK 1/2	0.937	2.65
QPSK 2/3	1.250	4.40
QPSK 3/4	1.406	5.30
16QAM 1/2	1.875	7.35
16QAM 2/3	2.500	10.10
16QAM 3/4	2.812	11.15
64QAM 2/3	3.749	14.70
64QAM 3/4	4.218	16.40
64QAM	5.624	21.35

Let us point out that the use of a given Modulation and Coding Scheme (MCS) level considering these thresholds provides some guarantees on maximum BER. As a first approximation, throughputs can actually be considered as goodput.

7.3.3 Step #3 - Allocating subchannels/PRBs

The third step allocates slots of resources to users thanks to a modified treemap algorithm. Since we consider wideband averaged channel, each user benefits the same from all frequency bands. So we do not pay

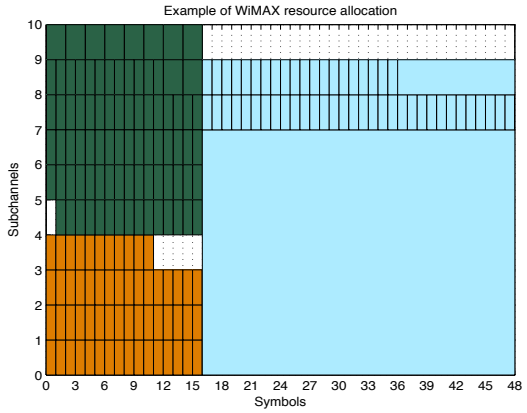
attention to which frequency band is allocated to which user, they are interchangeable.

The resource surface is first vertically divided between QoS vs non-QoS sessions. QoS sessions are scheduled first, and are then followed by non- QoS sessions. Because this separation must happen between symbols and not during a symbol, QoS sessions surface have to be ceiled, such as the division between QoS and non-QoS sessions happens after the $\left\lceil \frac{\sum_{u=1}^{N_{GBR}+N_{RT}} NT_u}{N_{subch}} \right\rceil$ symbol, where the operator $\lceil \cdot \rceil$ rounds to the closest larger integer. The amount of slots yet available for BE sessions is therefore reduced by this rounding waste (bounded by the number of subchannels/PRBs).

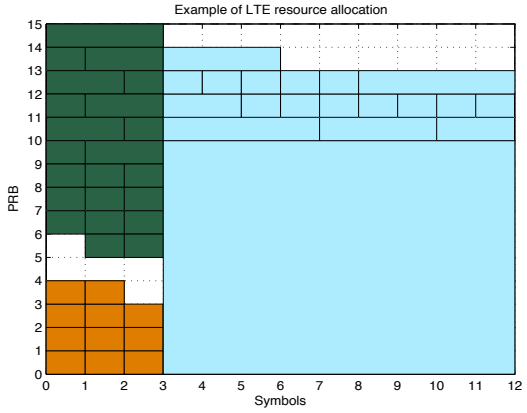
Then GBR sessions are separated from the RT ones, and finally resources are shared within each category. But for these last divisions, ceiling each user's need would cause a lot of resource waste. This is why we considered the stripe-based approach of the square treemap algorithm [68]. The idea is to draw a stripe as wide as one subchannel/PRB, and then divide that stripe according to the needs. Stripes are shared after each other, such as if needs of a session overwhelm one stripe, they can be spread on multiple stripes. To match the downlink constraint of rectangular shaped allocation, we can just consider, like [71] does, that when an allocation is spread over different stripes, they are actually different rectangle allocations given to the same user.

Figures 7.4(a) and 7.4(b) present a snapshot allocation we respectively obtained for WiMAX and LTE with our treemap-based resource allocation procedure. The first slots to be allocated are the GBR ones (in orange), then the RT ones (in green). They are allocated from horizontal stripes, starting from one side of the bandwidth for the GBR sessions, and from the other side for the RT ones. Then on the right hand side of the figures are the slots allocated for BE sessions (in blue). White slots are unused, because of rounding waste for the left hand side of the figures, or because BE sessions do not have anything more to transmit. The large blue square on the lower right hand side of the figures is actually allocated to a single BE session. Since BE sessions are sorted out according to a PF strategy, once a session is granted for transmission, it can receive as many resource slots as required to empty its queue.

As far as LTE resource allocation is concerned, let us point out that actually, PRB allocation is performed on a 0.5 ms slot basis, and not on a symbol scale. We however consider here a symbol scale allocation, to evaluate if this would make as much sense in LTE as it does in WiMAX.



(a) WiMAX



(b) LTE

Figure 7.4: Treemap resource allocation snapshot.

7.3.4 Algorithm

The scheduler can therefore be summarized by Algorithm 6. The first step sorts sessions out, according to the hierarchy described in Section 7.3.1. Then the second step evaluates the need of slots while there remains available slots. Finally, the third step maps slots to users.

The number $N_{available\ slots}$ is the amount of slots available for transmission. It is the product of the number of symbols per scheduling period N_{symb} by the number of subchannels/PRBs N_{subch}. The operator $\mathrm{mod}(\cdot, \cdot)$ is the modulo operation, such as it associates to $\mathrm{mod}(a, b)$ the remainder of the division $\frac{a}{b}$.

Algorithm 6 Treemap-based 3-step scheduling algorithm.

STEP #1:
Sort sessions according to the hierarchy described in Section 7.3.1

$$\Big\{U_{[1]},\ U_{[2]},\ \ldots, U_{[N_{GBR}]},\ \ldots, U_{[N_{GBR}+N_{RT}]},\ \ldots, U_{[N_{GBR}+N_{RT}+N_{BE}]}\Big\};$$

STEP #2:
$N_{available\ slots} \leftarrow N_{symb} \cdot N_{subch}$;
$k \leftarrow 1$; {current user for the loop}

while $N_{available\ slots} > 0$ **do**
Evaluate the capacity CS_k of a slot for the user U_k, according to TABLE 7.1 ;
Derive its required number of slots NT_k:

$$NT_k = \min\left\{\frac{Q_k}{CS_k}\ ,\ N_{available\ slots}\right\};$$

$N_{available\ slots} \leftarrow N_{available\ slots} - NT_k$;

if $k = N_{GBR}+N_{RT}$ **then** {Rounding for the cut after QoS sessions}
$N_{available\ slots} \leftarrow N_{available\ slots} - N_{subch} + \mod\left(\sum_{u=1}^{k} NT_u\ ,\ N_{subch}\right)$;
end if

$k \leftarrow k + 1$;
end while

STEP #3: Treemap resource allocation
Divide QoS vs non-QoS traffic after the $\left\lceil \frac{\sum_{u=1}^{N_{GBR}+N_{RT}} NT_u}{N_{subch}} \right\rceil$ symbol ;

Allocate slots to GBR traffic per stripe on the lower left corner;
Allocate slots to RT traffic per stripe on the upper left corner;
Allocate slots to BE traffic per stripe on the right hand side;

7.3.5 Complexity

The first step sorts users according to the hierarchy detailed in Section 7.3.1. Actually, GBR sessions do not need to be sorted since they all shall be scheduled. Because RT sessions are sorted according to an EDF strategy, the packets arrive already sorted in the queue. As far as BE sessions are concerned, we have to sort each of the three olympic types of sessions based on PF criteria. With a good sorting algorithm, this can be done in

$$\mathcal{O}\left(\max\{N_{BE_G}, N_{BE_S}, N_{BE_B}\} \cdot log\left(\max\{N_{BE_G}, N_{BE_S}, N_{BE_B}\} \right) \right),$$

where N_{BE_G}, N_{BE_S} and N_{BE_B} are respectively the number of Gold, Silver and Bronze BE sessions. Typically Bronze is the largest category. The computational complexity of this first step does not depend on the total amount of sessions.

At step #2, resource needs are assessed for sessions, one after another, according the hierarchy obtained at step #1. Only transmitting sessions have to be treated. They are limited by the number of slots ($N_{symb} \cdot N_{subch}$). Computational complexity of step #2 is therefore bounded by

$$\mathcal{O}\left(N_{symb} \cdot N_{subch}\right),$$

and does not depend on the number of sessions.

Finally, the complexity of step #3 depends on the treemap algorithm. The first division between QoS-constrained and BE sessions is immediate. Then slots are allocated linearly. As a result, complexity of step #3 is also bounded by

$$\mathcal{O}\left(N_{symb} \cdot N_{subch}\right).$$

Therefore, if there is a lot of BE sessions compared to the number of slots, the algorithm is dominated by the sorting algorithm of step #1. But if the cell handles a large amount of slots compared to the number of BE sessions, the computational complexity of the algorithm is linear with the number of slots.

To improve computational time, steps #1 and #2 should be done in parallel. Gold sessions should only be sorted once GBR and non-GBR RT session needs have been assessed, if there are still available slots. In the same way, Silver and Bronze sessions should only be sorted if there are still some available slots after respectively Gold and Silver sessions needs have been assessed. Indeed it is worthless to sort out BE sessions that can not be scheduled. We can save that time.

7.4 Handling uplink scenarios

Whereas DL transmissions of both WiMAX and LTE are based on OFDMA, some divergences arise with the UL side. While WiMAX uses OFDMA both on DL and UL, LTE prefers Single Carrier Frequency Division Multiple Access (SC-FDMA) for UL.

Indeed, OFDMA has the drawback of having a strongly varying signal envelope, causing a high Peak to Average Power Ratio (PAPR), since it transmits multiple data symbols in parallel, as [11] details. This causes high power consumption to decoding power amplifiers, which is an issue, especially for UEs in UL. To reduce PAPR, SC-FDMA uses a discrete Fourier transform to spread data symbols over the whole available bandwidth, over a single carrier frequency band [72]. As a result, symbols are therefore transmitted one after another, on a larger bandwidth at a faster pace. PAPR is therefore seriously reduced, greatly helping the energy consumption. On the receiver side, an inverse transform allows to retrieve transmitted symbols. However, because with SC-FDMA cyclic prefix is only added per bloc, there can be ISI within blocs, which makes receiver computations more complex, requiring more computational power. This therefore explains why LTE only selected SC-FDMA for UL, and not for both duplex sides.

Figure 7.5 compares the allocation of a single user (in grey) in OFDMA and SC-FDMA. In both cases, the user receives 4 subcarriers for a two-symbol duration, but information is sent differently. While with OFDMA information is transmitted in parallel over the 4 subcarriers, it is sequentially transmitted with SC-FDMA.

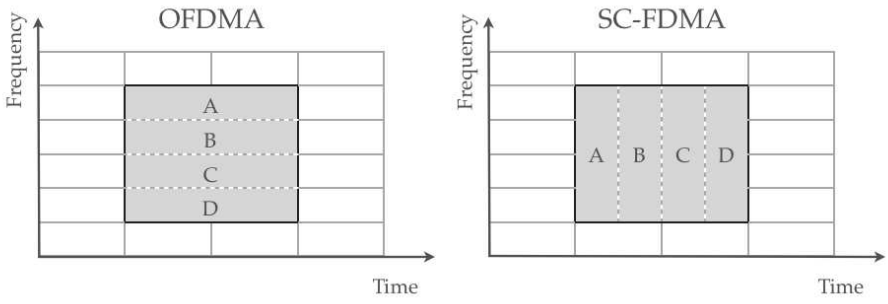


Figure 7.5: Comparison between OFDMA and SC-FDMA.

As far as resource allocation is concerned, the scheduler has to allocate rectangles of several subchannels/PRBs for a few symbol durations. So even if, due to different coding schemes, OFDMA and SC-FDMA do not provide the same capacity from the same amount of resource, our

scheduler does not have to consider if the allocated rectangles are going to be divided into parallel (OFDMA) or sequential (SC-FDMA) channels. It is therefore compatible with both UL multiple access techniques.

7.5 Two comparison algorithms

For comparison purpose, we implemented the algorithm HYGIENE presented in [52], and also a greedy algorithm derived from the MIMO algorithm of [42]. Besides providing some QoS guarantees, another interesting characteristic of these algorithms is that they both take advantage of frequency diversity, since they consider the effective quality of each subchannel/PRB instead of an averaged value. They are therefore able to schedule users on their best frequency bands. They should provide better cell throughput, but at the cost of a seriously increased feedback load. Another difference from our algorithm is that they allocate resources for a whole scheduling period duration. So allocation is simplified, but when users only require a fraction of a subchannel/PRB, some resources are wasted. As a result, comparison of the loss of frequency diversity gain vs wasted slots, on throughputs and packets delay aspects, should be interesting.

7.5.1 HYGIENE

HYGIENE [52] only distinguishes two kinds of sessions, the ones with a constant bit rate (CBR) to guarantee, and the BE ones. We match CBR sessions to our GBR and RT classes, while BE traffic is handled without considering user status.

HYGIENE first handles the set $\mathcal{K}_{\mathcal{CBR}}$ of CBR sessions, and iteratively allocates resources to the session k^* which has transmitted the least, such as

$$k^* = \underset{k \in \mathcal{K}_{\mathcal{CBR}}}{\arg\min} \left\{ \overline{C_{k,t}} \right\} \tag{7.1}$$

with $\overline{C_{k,t}}$ the achieved throughput of the session k from the beginning up to the current scheduling period t. The subchannel/PRB n^* which is going to be allocated to the session k^* is selected within the set $\mathcal{S}$ of subchannels/PRBs still available to be both the best one for the session k^* and the worst one on average for the set $\mathcal{K}_{\mathcal{BE}}$ of BE sessions:

$$n^* = \arg\min_{n} \left\{ \sum_{k \in \mathcal{K}_{\mathcal{BE}}} C_{\underset{n \in \mathcal{S}}{\arg\max}\left\{C_{n,k^*,t}\right\},k^*,t} \right\} \tag{7.2}$$

where $C_{n,k,t}$ is the channel capacity of the n^{th} Subchannel/PRB for the k^{th} session at the scheduling period t. As a result, though allocated before BE sessions, Constant Bit Rate (CBR) sessions should allow BE ones transmit on frequency bands of sufficiently good quality.

BE sessions are then iteratively scheduled according to a PF strategy. The BE session k^* gets its best Subchannel/PRB n^* such as:

$$(n^*,\, k^*) = \underset{n \in \mathcal{S}, k \in \mathcal{K}_{\mathcal{BE}}}{\arg\max} \left\{ \frac{C_{n,k,t}}{\overline{C_{k,t}}} \right\} \tag{7.3}$$

Let us point out that once a Subchannel/PRB has been allocated to a user, it is allocated for a whole TTI length, even if the session only requires it for a few symbols duration. As far as computational complexity is concerned, this algorithm is $\mathcal{O}(N_{subch}^2 \cdot N_{queued})$, which is much higher than the complexity of our scheme.

7.5.2 An iterative greedy algorithm

As a second reference algorithm, we implemented a simplified SISO version of the MIMO iterative greedy algorithm proposed by [42]. It also tries to maximize the channel capacity, taking into account the channel response of each subchannel/PRB. Each user therefore has to feed back CQI for each subchannel/PRB.

The authors of [42] define as a *stream* (k, n) a pair of a user U_k and a subchannel/PRB n. Greedily, they look for the best set of streams $\mathcal{P}^*$ that maximizes the weighted cell throughput $\mathcal{R}$:

$$\mathcal{R}\left(\mathcal{P}\right) = \sum_{(k,n) \in \mathcal{P}} w_k \cdot C_{n,k} \tag{7.4}$$

To provide some priority to QoS-guaranteed sessions, instead of the PF weights proposed originally in [42], which would have introduced fairness between sessions of different QoS categories, we chose

$$w_k = \begin{cases} 4 & \text{if } U_k \text{ is a GBR session} \\ 2 & \text{if } U_k \text{ is a RT session} \\ 1 & \text{if } U_k \text{ is a BE session} \end{cases} \tag{7.5}$$

As in [42], and like all other considered algorithms, uniformly distributed power has been used. Computational complexity of this algorithm is $\mathcal{O}(N_{subch} \cdot N_{queued})$, which is also higher than our scheme, though lower than HYGIENE.

7.6 Traffic generation

We model three kinds of sessions, namely GBR, RT and BE. Unless mentioned otherwise, the amount of sessions we consider is summarized in Table 7.2. The proportion of each kind of traffic has been derived from [16, p. 53 and 73], to reflect realistic traffic distributions.

Table 7.2: Traffic distribution of our simulations.

Traffic type	Model of traffic	Number of sessions
GBR	VoIP	140
RT	Video streaming	120
BE Gold	HTTP	20
BE Silver	HTTP	40
BE Bronze	HTTP	120

7.6.1 GBR traffic model

GBR sessions are modeled as VoIP traffic, like [16, p. 26] details. They represent a mean throughput of 12.5kbps each. Those throughputs are obtained from the transmission of 32-Byte packets every 20ms when sessions are active. The session activity is determined through a two state Markov process, with a probability to remain active of 0.99, and a probability to remain inactive of 0.9875.

7.6.2 RT traffic model

RT sessions are modeled according to the video streaming model of [73, p. 62]. Their mean throughput is around 50kbps. Flows consist of 100ms video frames, divided into 8 slices. The size of these slices follows a Pareto distribution truncated between 40 and 250 Bytes. The time interval between slices also follows a Pareto distribution, truncated between 2.5ms and 12.5ms. The alpha parameter of these distributions is $\alpha = 1.2$.

We did not introduced any network latency. The EDF scheduling of the RT packets is therefore based on the time packets spent in buffer queues. We can however consider that random variations in time interval between packets take into account some network latency variations, though they are not large enough to modify packets order.

7.6.3 BE traffic model

BE model is derived from the HTTP model of [73, p. 59], with the same parameters for Gold, Silver and Bronze users. Web pages are composed of a main object followed by several attached objects. Objects' size follow a lognormal distribution, truncated between 100 bytes and 2Mb with parameters $\sigma = 1.37$ and $\mu = 8.35$ for the main objects, or truncated between 50 bytes and 2Mb with parameters $\sigma = 2.36$ and $\mu = 6.17$ for the attached objects. The attached objects follow their main object after a parsing time, which follows an exponential distribution of 130ms mean. The reading time of a page, before asking another one, also follows an exponential distribution, but of 30s mean.

7.7 Validation of our implementations

To ensure the correctness of our scheduling algorithms implementations, we tested the mixed GBR/non-GBR LTE scenario of [2]. In similar conditions, Fig. 7.6 shows the cumulative probability distribution of GBR and BE session throughputs for our algorithm (TM) and for the reference algorithms HYGIENE and the iterative greedy algorithm (Greedy). Queue rates (QR) are also plotted since they act as bounds on throughputs, since sessions can not transmit more than the content of their queue. All the simulation parameters are summarized within Table 7.3.

In their scenario, the authors of [2] managed to schedule 300+ VoIP sessions along with 200 128-kbps BE sessions. As one can see on Fig. 7.6, we are able to serve 390 GBR sessions while 20% of the 200 on-going BE ones have a throughput higher than 128kbps. Actually, this limitation in the BE throughputs comes from the traffic generation model. As queue rate curves show, only 25% of the BE sessions can have a throughput higher than 128kbps. As far as reference algorithms are concerned, HYGIENE curves are quite similar to ours, while the greedy algorithm seems quite less efficient.

HYGIENE and our algorithm seem to have close curves, but their BE mean throughputs are quite different. Actually, the mean throughput for our algorithm is higher because of extreme values. Only 1% of the BE sessions have a throughput larger than 1,000kbps for the reference algorithm, compared to 5% for our algorithm. Indeed, HYGIENE picks a BE session for each subchannel/PRB, while our algorithm picks one BE session, and tries to empty its queue before picking another BE session.

Parameter	Value
Number of users	390 GBR + 200 BE
Traffic model	GBR: VoIP BE: HTTP
Channel model	Temporally correlated Rayleigh
Bandwidth	50 PRBs (9MHz)
Carrier Frequency	2GHz
Resource to share	Channel slots
User location	Between 35 and 500m away from access point
User velocity	30kmph
Transmit power	32W
Noise power	-101.2 dBm
Interference model	No interference
Antenna pattern	SISO
TTI duration	1ms
Run duration	100 TTIs
Number of runs	100 runs

Table 7.3: Parameters for OFDMA SISO validation.

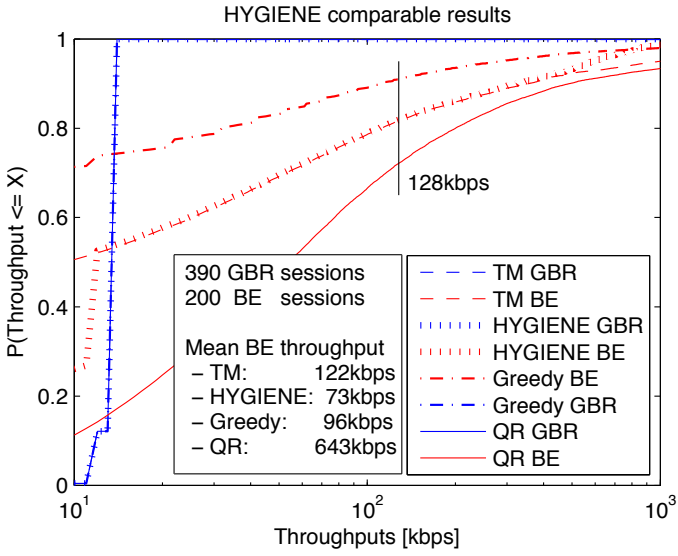


Figure 7.6: Cumulative probability distribution of session throughputs in a scenario comparable to [2].

7.8 A channel capacity bound

The authors of [52] have also presented an upper bound on channel capacity. But it can not handle packetized traffic, and computations are quite heavy since the allocation for all scheduling periods are computed altogether. To benchmark our algorithm with some optimal results, we designed a slightly different model of bound, handling packetized traffic, providing QoS, and for computational time efficiency, allocating resources on a per TTI/frame basis.

This bound evaluation is therefore looking for the $\rho_{k,n,t} \in [0,1]$, which are the fractions of the n^{th} subchannel/PRB allocated to the session k during the t^{th} scheduling period, for all sessions, subchannels/PRBs and frames/TTIs.

For each frame/TTI denoted by the subscript T, we have to find

$$\rho^*_{k,n,T} = \arg\max \sum_{k=1}^{N_{users}} \sum_{n=1}^{N_{subch}} C_{k,n,T} \cdot \rho_{k,n,T} \cdot w_{k,T} \tag{7.6}$$

under these conditions:

$$\sum_{k=1}^{N_{users}} \rho_{k,n,T} \leq 1 \qquad \forall n = 1, \ldots, N_{subch}; \tag{7.7a}$$

$$\sum_{n=1}^{N_{subch}} \sum_{t=1}^{T} C_{k,n,t} \cdot \rho_{k,n,t} = \sum_{t=1}^{T} Q_{k,t} \qquad \forall k \in GBR; \tag{7.7b}$$

$$\sum_{n=1}^{N_{subch}} \sum_{t=1}^{T} C_{k,n,t} \cdot \rho_{k,n,t} \leq \sum_{t=1}^{T} Q_{k,t} \qquad \forall k \in RT, BE; \tag{7.7c}$$

$$\sum_{n=1}^{N_{subch}} \sum_{t=1}^{T} C_{k,n,t} \cdot \rho_{k,n,t} \geq \sum_{t=1}^{T-T_m} Q_{k,t} \qquad \forall k \in RT. \tag{7.7d}$$

Each $w_{k,T}$ coefficient is the backlog of session k until the T^{th} scheduling period. These weights help fairness between users. Condition (7.7a) prevents us from allocating more than 100% of each subchannel/PRB. GBR sessions have their traffic guaranteed by (7.7b), while (7.7c) only prevent RT and BE sessions from transmitting more than what is in their queues. Condition (7.7d) prevents RT packets from remaining longer than T_m scheduling periods in the queue, providing some maximal delay guarantee.

For each scheduling period, this is a Linear Programming problem we solved with the SIMPLEX algorithm implementation of MATLAB. Let us mention that this is an upper bound on the total cell capacity,

providing some QoS and fairness constraints. This is not a bound on each session's throughput. Let us also point out that frequency bands are not fractioned at symbol scale, so results are overestimated.

7.9 DL WiMAX evaluation

To evaluate the performances of our treemap-based algorithm, we implemented it into MATLAB scripts, and simulated first the behavior of DL WiMAX environment. Each user's feedback is limited to 5bits per scheduling period, as in [57]. This means each user only requires 1kbps of feedback, whatever the total cell bandwidth is.

We compared throughputs and packet delays for our treemap-based algorithm with those of the reference algorithms described in Section 7.5 and those of our bound of Section 7.8. Following results are 50% and 95% outage probabilities, which means that respectively 50% or 95% of the users have at least the throughput or delay described as outage.

Parameter	Value
Number of users	140 GBR + 120 RT + 180 BE
Traffic model	GBR: VoIP RT: Video streaming BE: HTTP
Channel model	Temporally correlated Rayleigh
Bandwidth	10 - 100 subchannels
Carrier Frequency	2GHz
Resource to share	Channel slots
User location	Uniformly distributed within a 120° sector cell (2.8km between BS)
User velocity	30kmph
Transmit power	20W
Noise power	-101.2 dBm
Interference model	No interference
Antenna pattern	SISO
TTI duration	5ms
Run duration	100 TTIs
Number of runs	100 runs (10 runs for bound evaluations)

Table 7.4: Parameters for SISO DL WiMAX computations.

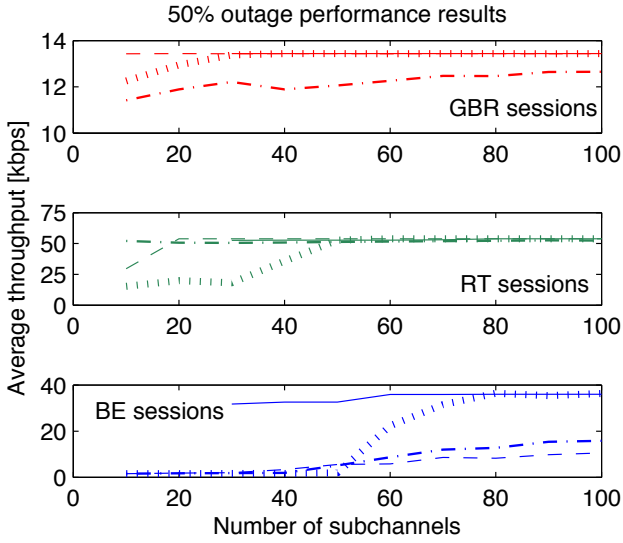
We consider a radio channel impaired with temporally correlated Raleigh fading, independent per subcarrier. Users are uniformly distributed within a 120° sector of a hexagonal cell, considering a distance between base stations of 2.8km. CQI values have been derived from [74], with a pathloss exponent of 4.0, a Doppler shift based on 30kmph of user velocity, and a carrier frequency of 2GHz. Results are obtained considering 10 to 100 subchannels, with the sessions distribution pattern described by Table 7.2. All the simulation parameters are summarized within Table 7.4.

Figure 7.7 presents first the outage throughputs of each traffic category according to the number of available subchannels for our algorithm, the reference algorithms and our bound. At both 50% and 95% outage, our treemap-based algorithm is able to provide guaranteed bit rate to GBR sessions, even with only 10 subchannels, whereas HYGIENE requires at least 40 subchannels for the same performance results. HYGIENE also requires more subchannels to provide enough throughput to RT sessions, namely 50+, compared to only 30 subchannels for our scheduler. The greedy algorithm produces quite lower GBR throughputs, but provides RT throughputs larger than what the other two algorithms produce. This can be explained by its priority system based on weights, which is not a system as strict as the hierarchy of the other two algorithms.

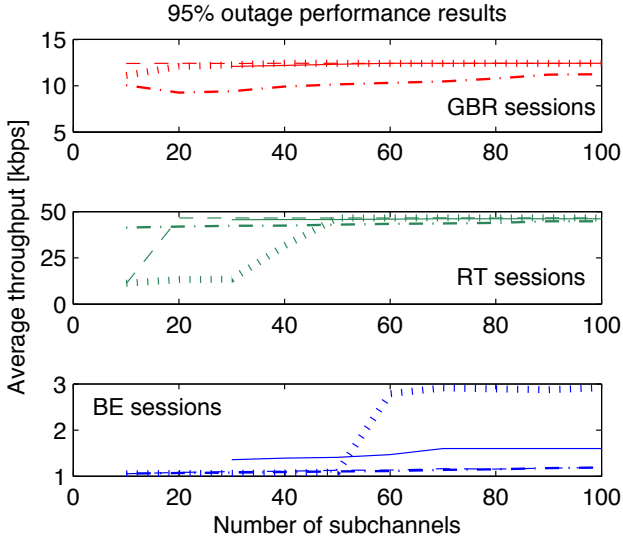
When it comes to BE traffic, when there are enough subchannels, HYGIENE provides higher throughputs than our scheduler, and even higher than the bound at 95% outage. The bound is actually a bound on cell throughputs, and is not as fair as the reference algorithm. Indeed, it limits resources allocated to users with poor channel, to maximize total throughput, therefore reducing the 95% outage results. BE results of the greedy algorithm are quite similar to ours. Let us also point out that with only 10 or 20 subchannels, the optimization problem of the bound encounters unfeasibility issues. This is the reason the bound throughput curves only begin with 30 subchannels.

Results of Fig. 7.7 may seem quite low with respect to those of Fig. 7.6. Let us recall that Fig. 7.7 presents the minimum throughputs that 50% or 95% of users benefit from. BE throughputs measured on Fig. 7.6 at 95% outage also represent only a few kbps. Moreover, we considered here a base station to base station distance of 2.8km, instead of the maximum 500m of user to base station distance of [2].

Figure 7.8 shows the total cell throughput at both 50% and 95% outage probability. For limited bandwidth, our algorithm outperforms the reference ones, especially HYGIENE. Indeed, because the reference algorithms allocate subchannels for a whole frame duration to users,

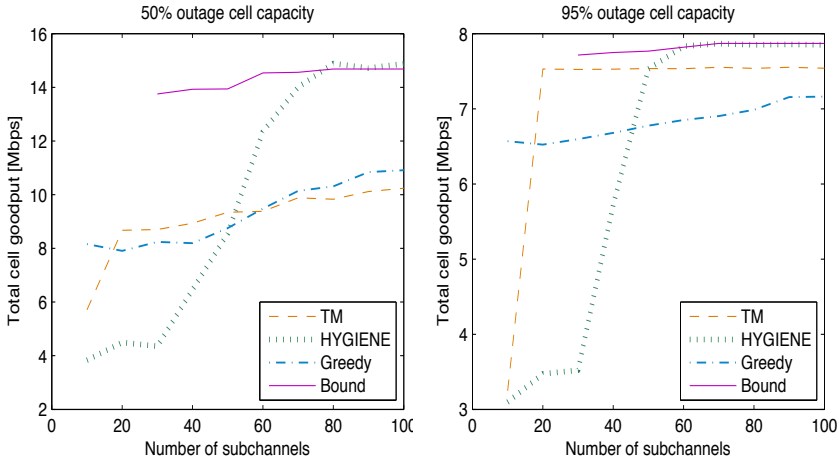


(a) 50% outage probability



(b) 95% outage probability

Figure 7.7: DL WiMAX outage throughputs. Dashed lines stand for our scheme, dotted ones for HYGIENE, dash-dotted ones for the reference algorithm and solid ones for the bound.



(a) 50% outage probability (b) 95% outage probability

Figure 7.8: DL WiMAX outage cell throughputs.

some resources are wasted. It is only when bandwidth is sufficient that reference algorithms manage to provide a better throughput than our scheduler. HYGIENE matches the bound at 50% outage with 80 subchannels, or even at 95% outage with only 60 subchannels. Because our treemap-based scheduler does not benefit from frequency diversity, which is improved by the augmentation of bandwidth, and instead only considers an averaged channel, its throughput is not as increased as the reference one is when bandwidth is widened.

Let us also consider the feedback cost. Figure 7.9 shows the relative throughput gain of the reference algorithms over our treemap-based scheme, and compares it to the relative increase of feedback (FB) they require, at both 50% and 95% outage probability. The reference gain Ref_{gain} and the feedback overload $FB_{overload}$ are defined as follows

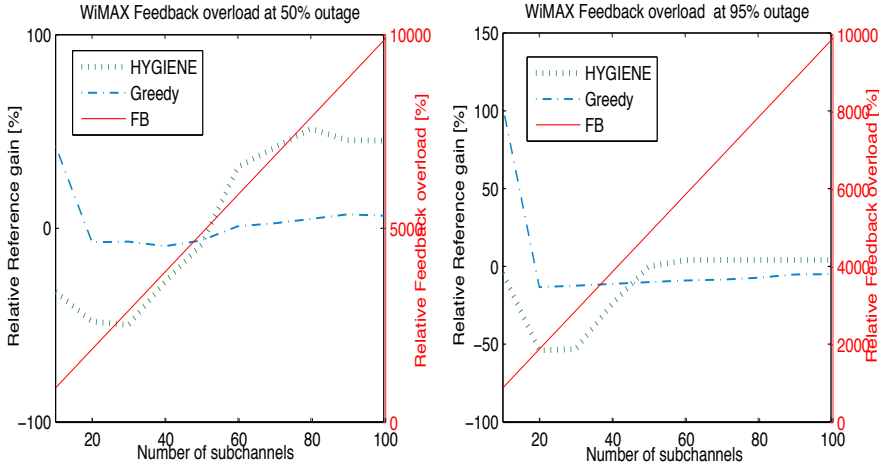
$$Ref_{gain} = \frac{C_{ref} - C_{TM}}{C_{TM}} \tag{7.8}$$

$$FB_{overload} = \frac{FB_{ref} - FB_{TM}}{FB_{TM}} \tag{7.9}$$

where C_{ref}, C_{TM}, FB_{ref} and FB_{TM} are respectively the throughput obtained by one reference algorithm, the throughput obtained by our treemap-based algorithm, the feedback load required by one reference algorithm and the feedback load required by our algorithm.

As one can see in Fig. 7.9, at low bandwidth, the HYGIENE gain is actually negative. HYGIENE only becomes better than our scheme when

there are more than 50 subchannels, but at the price of a tremendous feedback load. Indeed, feeding back 5bits for each of the 50 subchannels every 5ms frame leads to 50kbps of feedback per user, which is the average throughput of our modeled RT sessions. As for the greedy algorithm, it produces no significative gain on our algorithm, but has the same feedback cost as HYGIENE.

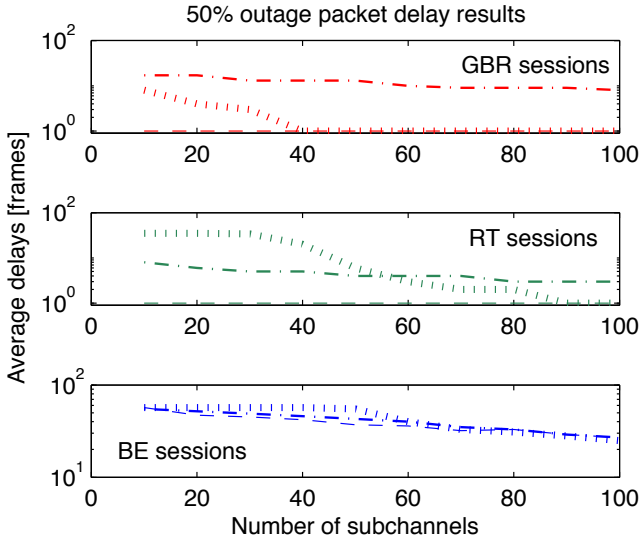


(a) 50% outage probability

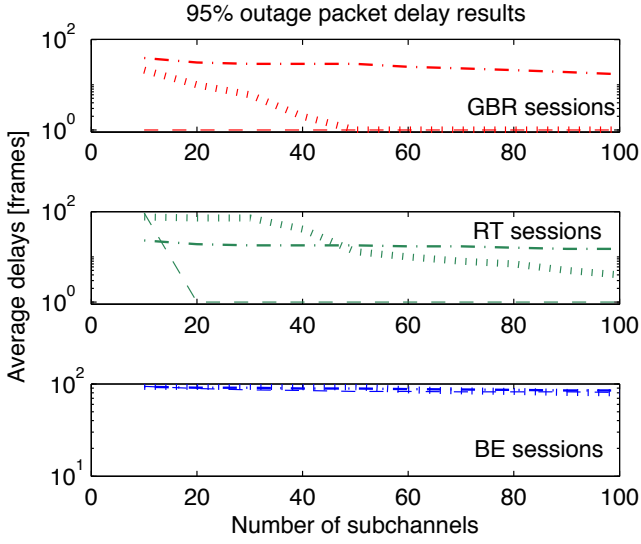
(b) 95% outage probability

Figure 7.9: DL WiMAX reference outage throughput gain on our scheme, compared to the feedback cost.

As far as packet delays are concerned, Fig. 7.10 presents for each traffic category the outage delays, expressed in 5ms frames. Because our algorithm starts scheduling with GBR packets, this kind of traffic is not delayed, even when there are only 10 subchannels. For similar performance, HYGIENE, which does not distinguish GBR from RT sessions, requires at least 40 subchannels at 50% outage, or even 60 subchannels at 95% outage probability. Our scheduler does not delay RT traffic either, provided there are more than 20-30 subchannels, whereas HYGIENE requires at least 50-60 subchannels for less than a 10-frame delay. As for the greedy algorithm, which only uses weights to prioritize traffic, it produces quite large delays for GBR while providing better RT delays than HYGIENE but still not as good as ours. As far as BE packets are concerned, they undergo large delays, with all algorithms. Even with 80 subchannels, half of the packets have to wait at least 20 frames. And 5% of these packets have to wait more than 80 frames, even with 100 subchannels. Actually, cells are so overloaded that most of these packets are never transferred within our 100-frame simulations.



(a) 50% outage probability



(b) 95% outage probability

Figure 7.10: DL WiMAX packet delay results. Dashed lines stand for our scheme, dotted ones for HYGIENE and dash-dotted ones for the greedy algorithm.

We also considered fairness between users. We evaluated it with the Jain Index [23] that we already defined in Section 3.4.5. Figure 7.11 presents the Jain Index for a growing number of subchannels for RT and BE traffics. Since GBR packets have to be transmitted as soon as possible, it is not relevant to discuss about fairness for this kind of traffic. For RT traffic, with only 10 subchannels, the greedy algorithm is the fairest, but with so few resources, its throughputs are really insufficient. With 10 more subchannels, our algorithm manages to become as fair as the greedy algorithm. From 40 subchannels on, all algorithms become as fair as each other, and actually fully serve each RT session.

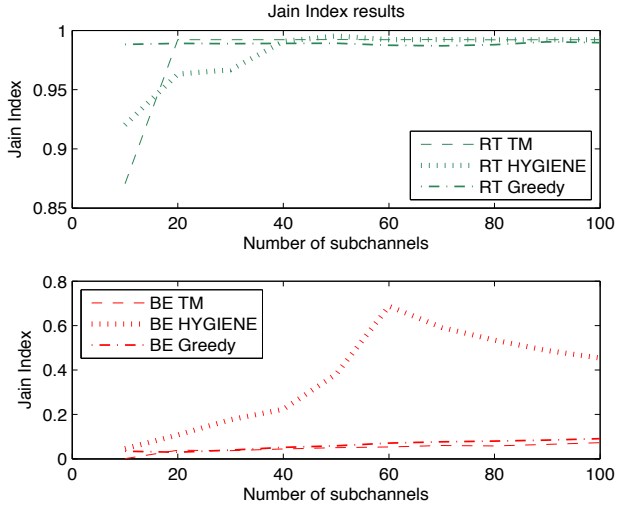


Figure 7.11: DL WiMAX fairness for RT and BE sessions. Dashed lines stand for our scheme, dotted ones for HYGIENE and dash-dotted ones for the greedy algorithm.

As far as BE traffic is concerned, HYGIENE becomes fairer and fairer as the number of available subchannels grows, until 60 subchannels. Then with such a bandwidth, GBR and RT sessions are fully served by the reference algorithm, and fairness between BE sessions decreases. Fairness between BE sessions produced by both our algorithm and the greedy one is really poor. Actually, for each scheduling period, when our scheduler selects a BE session for transmission, it tries to empty its queue, and then only if there are still available resources, tries to find another BE session, whereas HYGIENE selects another session for each subchannel. Moreover, our algorithm differentiates Gold, Silver and Bronze profiles, whereas HYGIENE does not. Poor fairness of the greedy algorithms directly comes from its maximum weighted throughput criteria.

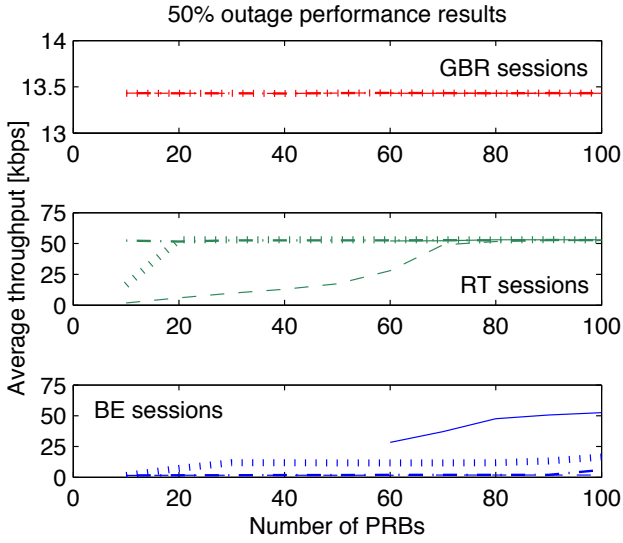
7.10 DL LTE evaluation

To evaluate the influence of a shorter scheduling period, we considered some DL LTE scenarios, with 1ms TTIs instead of 5ms frames. Reference algorithms' resource wastes, due to frequency bands allocation to users for whole scheduling periods, should therefore no longer be an issue. To better evaluate the influence of the shorter scheduling period, we still considered a per-symbol allocation as in WiMAX, though LTE allocation is actually performed on a 0.5ms slot basis. Let us also point out that LTE PRBs represent a narrower bandwidth than WiMAX subchannels, and 100 PRBs only correspond to 70 subchannels. Other simulation parameters are the same as in our DL WiMAX simulations, with the sessions distribution described in Table 7.2. All the simulation parameters are summarized within Table 7.5.

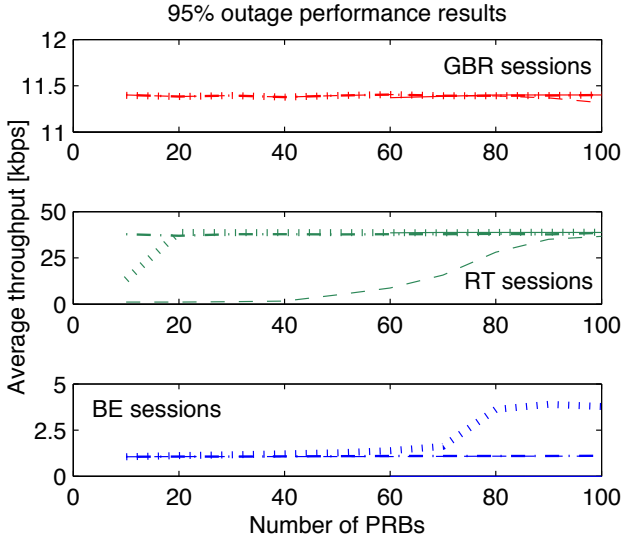
Parameter	Value
Number of users	140 GBR + 120 RT + 180 BE
Traffic model	GBR: VoIP RT: Video streaming BE: HTTP
Channel model	Temporally correlated Rayleigh
Bandwidth	10 - 100 PRBs
Carrier Frequency	2GHz
Resource to share	Channel slots
User location	Uniformly distributed within a 120° sector cell (3km between BS)
User velocity	30kmph
Transmit power	32W
Noise power	-101.2 dBm
Interference model	No interference
Antenna pattern	SISO
TTI duration	1ms
Run duration	100 TTIs
Number of runs	100 runs (10 runs for bound evaluations)

Table 7.5: Parameters for SISO DL LTE computations.

Figure 7.12 first presents the outage throughput results. All algorithms manage to provide guaranteed QoS to GBR flows, but as far as RT and BE sessions are concerned, the reference algorithms provide much better results. The evaluation of the bound also encounters unfea-



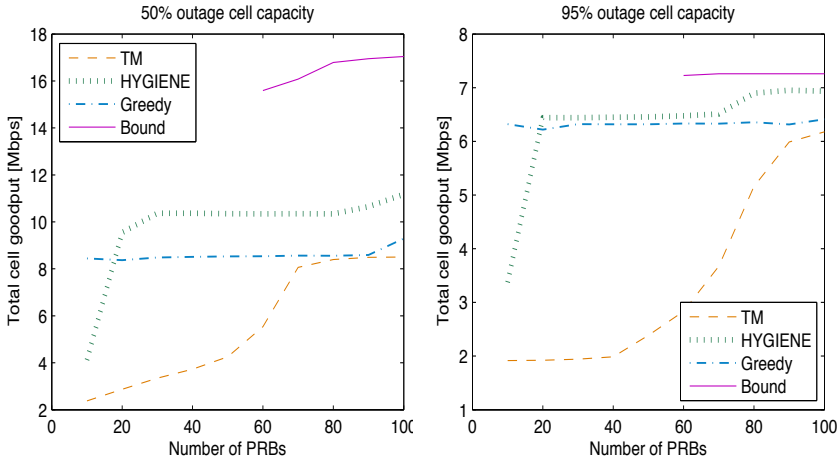
(a) 50% outage probability



(b) 95% outage probability

Figure 7.12: DL LTE outage throughputs. Dashed lines stand for our scheme, dotted ones for HYGIENE, dash-dotted for the greedy algorithm, and solid ones for the bound.

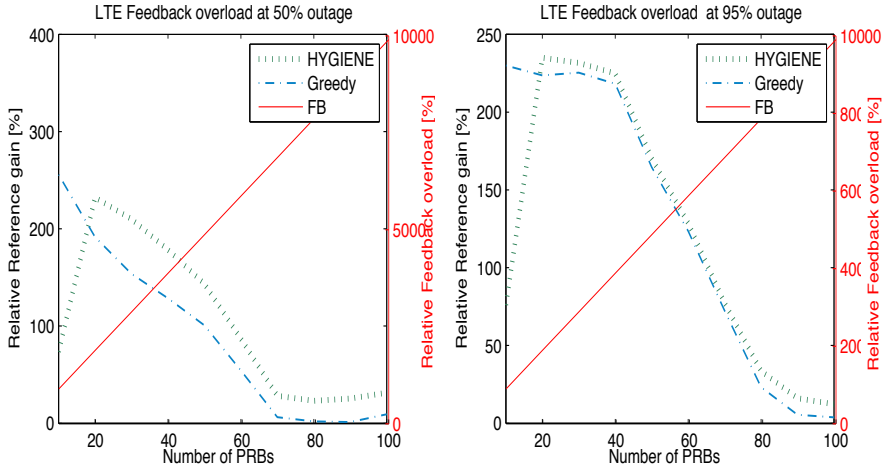
sibility issues, when there are less than 50-60 PRBs, which correspond to a 30-subchannel bandwidth.



(a) 50% outage probability (b) 95% outage probability

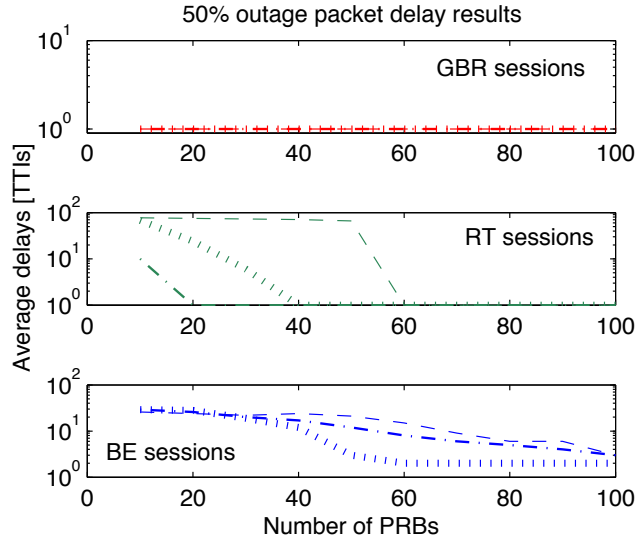
Figure 7.13: DL LTE outage cell throughputs.

Outage cell throughput results of Fig. 7.13 also point out a better throughput for both reference algorithms, especially for HYGIENE, though significantly lower than the bound.

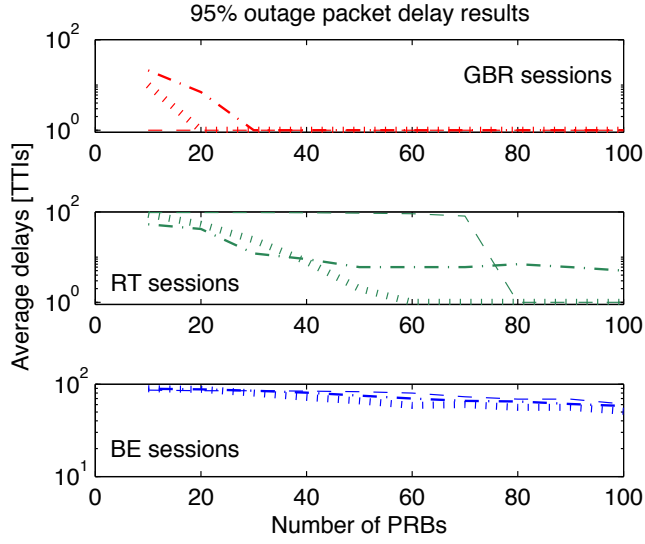


(a) 50% outage probability (b) 95% outage probability

Figure 7.14: DL LTE reference outage throughput gain on our scheme, compared to the feedback cost.



(a) 50% outage probability



(b) 95% outage probability

Figure 7.15: DL LTE packet delay results. Dashed lines stand for our scheme, and dotted ones for the reference algorithm.

As far as relative reference gain is concerned, as one can see on Fig. 7.14, both reference algorithms provide quite similar gain. They are much better when there are only a few PRBs, but as the number of PRBs grows, this advantage disappears, while feedback load seriously increases.

The packet delays for a growing number of PRBs are shown by Fig. 7.15. From 20-30 PRBs, all algorithms are able to deliver GBR packets with no additional delay. RT sessions are better served by the reference algorithms, but as far as BE traffic is concerned, results are quite poor, like those obtained with WiMAX.

Finally, the Jain Index fairness results are presented by Fig. 7.16. The reference algorithms are quite fair between RT flows, while our algorithm requires 80 PRBs to be as fair. We obtain the same fairness behavior for BE sessions as in WiMAX. HYGIENE fairness increases until 30 PRBs, then progressively decreases, whereas fairnesses of both our algorithm and the greedy one remain quite low.

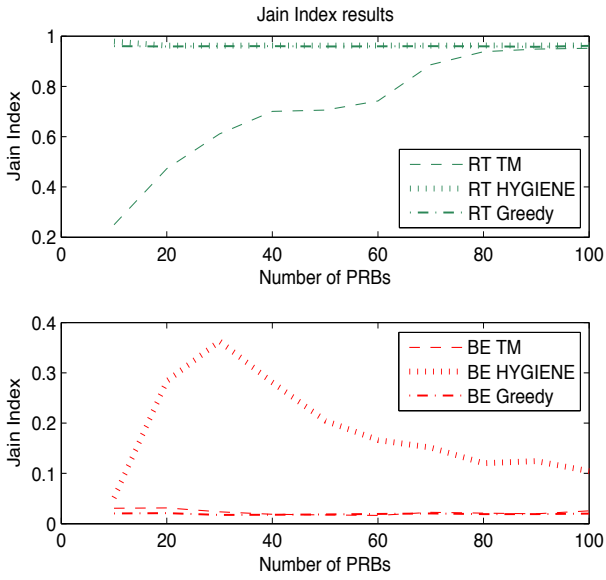


Figure 7.16: DL LTE fairness for RT and BE sessions. Dashed lines stand for our scheme, and dotted ones for the reference algorithm.

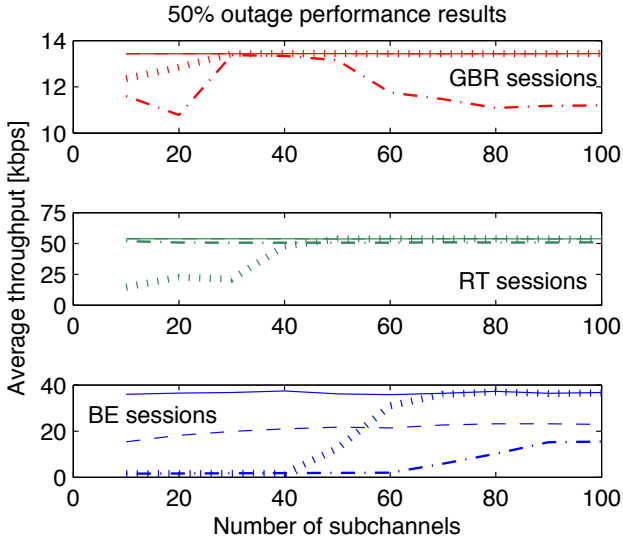
7.11 UL WiMAX evaluation

We finally considered UL scenarios, and more precisely some FDD Mobile WiMAX ones. While in DL scenarios the transmission is centralized, in UL ones, it is the distant users which transmit simultaneously. In DL we considered a total cell transmit power of 20W in WiMAX or 32W in LTE, shared among the varying number of subchannels/PRBs. So the more subchannels/PRBs are handled, the less each of them receives power. In UL, the transmit power used for each frequency band depends on the power of its user. Here we consider a fixed transmit power of 200mW per subchannel. This is the specific part of these UL results, other parameters are the same as in our DL WiMAX simulations, with the same sessions' distribution of Table 7.2. All the simulation parameters are summarized within Table 7.6.

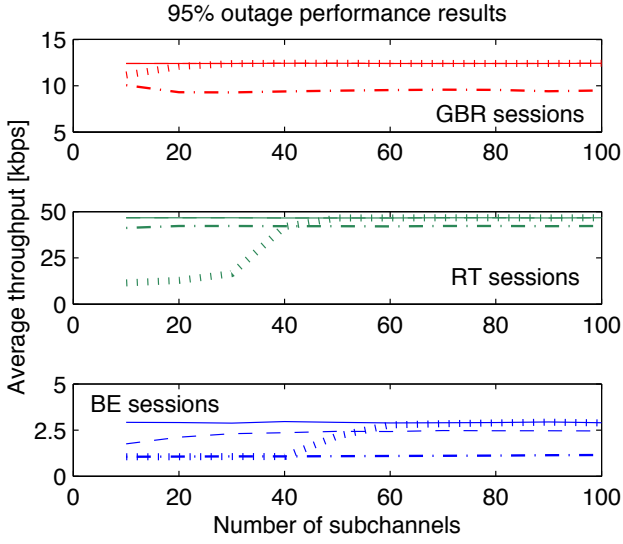
Parameter	Value
Number of users	140 GBR + 120 RT + 180 BE
Traffic model	GBR: VoIP RT: Video streaming BE: HTTP
Channel model	Temporally correlated Rayleigh
Bandwidth	10 - 100 subchannels
Carrier Frequency	2GHz
Resource to share	Channel slots
User location	Uniformly distributed within a 120° sector cell (2.8km between BS)
User velocity	30kmph
Transmit power	200mW per subchannel
Noise power	-101.2 dBm
Interference model	No interference
Antenna pattern	SISO
TTI duration	5ms
Run duration	100 TTIs
Number of runs	100 runs

Table 7.6: Parameters for SISO UL WiMAX computations.

Figure 7.17 presents the throughput evolution for a growing number of subchannels for all algorithms, and also for the queue rates. Since users can not transmit more data than what is in their queue, queue rate acts as a bound on throughput.



(a) 50% outage probability

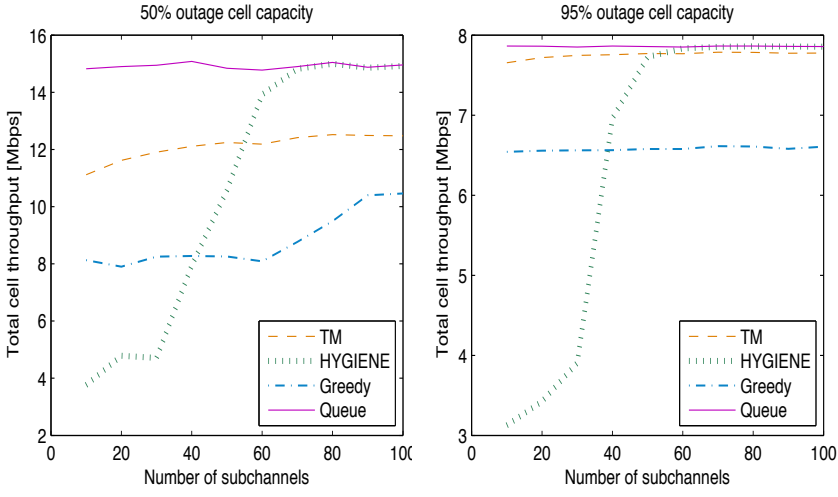


(b) 95% outage probability

Figure 7.17: UL WiMAX outage throughputs. Dashed lines stand for our scheme, dotted ones for HYGIENE, dash-dotted ones for the greedy algorithm, and solid ones for the queue rates.

As in the DL WiMAX scenario, our treemap-based scheduler is able to fully serve GBR and RT sessions, even with only 10 subchannels, while HYGIENE requires at least 30 subchannels for GBR, and 40 subchannels for RT traffic. As for the greedy algorithm, its GBR results are quite poor, though it nearly manages to fully serve RT traffic. BE sessions are also better served by our algorithm. With only a few subchannels, the reference algorithms, already struggling to let QoS guaranteed sessions transmit, can not let any BE session transmit. However when there are more subchannels, and that QoS-guaranteed sessions are sufficiently served, HYGIENE manages to better serve BE sessions.

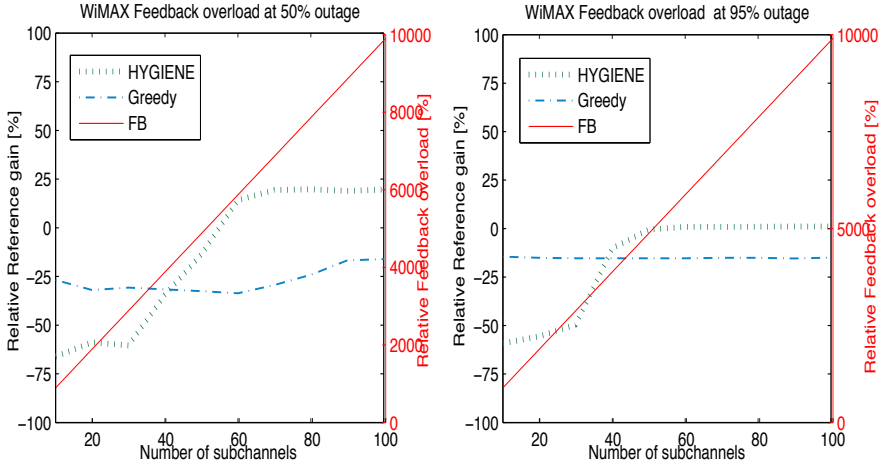
Those differences appear more evidently on the outage cell throughput results of Fig. 7.18. When there are only a few subchannels, our algorithm produces much better throughput than the greedy algorithm and HYGIENE. This is because our algorithm can share subchannels to several users on a symbol scale while reference algorithms allocate subchannels to a single user for a whole frame duration. But like in DL, as bandwidth is increased, HYGIENE outperforms other schedulers, and matches the total queue rate when there are 50+ subchannels. As far as the greedy algorithm is concerned, its cell throughput always remains lower than our algorithm's.



(a) 50% outage probability (b) 95% outage probability

Figure 7.18: UL WiMAX outage cell throughputs.

Reference outage throughput gains of Fig. 7.19 also point out these behaviors. The greedy algorithm gain is always negative, while HYGIENE produces some gain when there are more than 50 subchannels. But this throughput gain is a bit low compared to the reference feedback requirement, 50 to 100 times more important for 50 to 100 subchannels than ours.



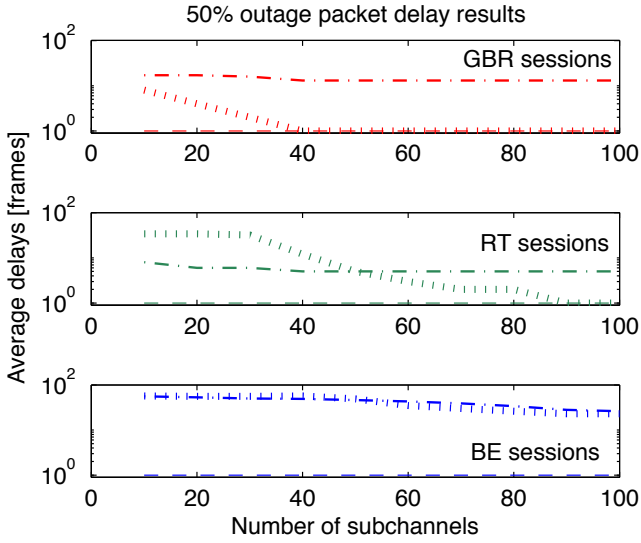
(a) 50% outage probability

(b) 95% outage probability

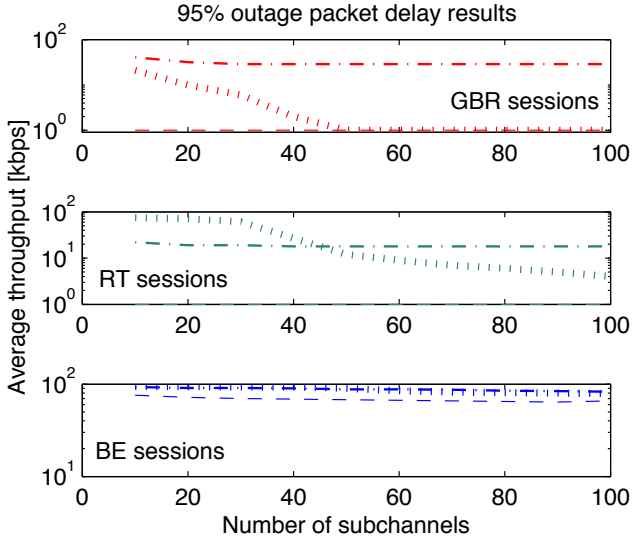
Figure 7.19: UL WiMAX reference outage throughput gain on our scheme, compared to the feedback cost.

Packet delay results are shown at Fig. 7.20. Our algorithm is able to serve without delay GBR and RT sessions. Delays obtained by the reference algorithms are much poorer. HYGIENE requires at least 40 subchannels to transmit without delays GBR packets, while RT packets have to wait a dozen frames, and the greedy algorithm always delays the GBR and RT packets a lot, whatever the available amount of subchannel is. BE delays remain quite high, for all algorithms and number of available subchannels. Actually some BE packets can not be transmitted at all during the simulation time, BE queues can not be emptied.

Fairness results are presented at Fig. 7.21, and here in UL, it is our algorithm which is the fairest for RT traffic, but quite close to the greedy algorithm. After 40 subchannels, all algorithms provide the same fairness. As far as BE traffic is concerned, HYGIENE's fairness is still increasing, up to the point where QoS-guaranteed sessions are sufficiently served, then it is decreasing. Here in UL, our algorithm seems fairer with BE sessions, and even fairer than HYGIENE below 40 subchannels, while the greedy algorithm remains quite unfair.



(a) 50% outage probability



(b) 95% outage probability

Figure 7.20: UL WiMAX packet delay results. Dashed lines stand for our scheme, and dotted ones for the reference algorithm.

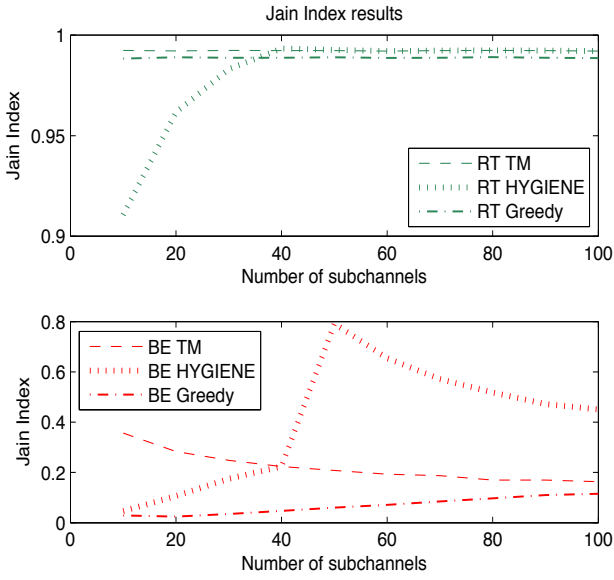


Figure 7.21: UL WiMAX fairness for RT and BE sessions. Dashed lines stand for our scheme, and dotted ones for the reference algorithm.

7.12 Concluding remarks

We have presented a hierarchical scheduling heuristic based on treemap visualization algorithms, greatly reducing feedback load requirements. It first sorts out sessions according to their QoS constraints, then assesses their requirements in resource slots. Thereafter a modified treemap algorithm allocates these resource slots to sessions. Our algorithm is quite simple and provides QoS guarantees, even within overloaded cells.

We implemented our algorithm into MATLAB scripts, and compared it to two reference algorithms (HYGIENE [52], and a SISO implementation of the greedy algorithm of [42]), and to a bound that we set up on channel capacity. We performed computer simulations to evaluate the performance of our scheduler in DL WiMAX, DL LTE and UL WiMAX frameworks.

With WiMAX, both in DL and UL, our algorithm provides better throughputs than the reference algorithms when subchannels are scarce. It even manages to empty GBR and RT queues, and reach their throughput bounds. This leads to better throughputs, and shorter packet delays.

These performance can be explained by the resource allocation scheme. While both reference algorithms allocate frequency bands per scheduling period, our algorithm only allocates them for as many symbols as each user needs. Our algorithm prevents some resources to be wasted, and is therefore able to schedule more users per frame/TTI. As a result, users can be scheduled more often, which reduces delays. But with a large amount of subchannels, more than 50 with our parameters, the frequency selectivity HYGIENE benefits from enables it to provide a higher channel capacity than our algorithm, which only considers an averaged wideband CQI feedback scheme. These performance are obtained at the price of a tremendous feedback load. To handle 50 subchannels, the reference scheduler requires a 50-kbps feedback from each user, whereas our algorithm only requires 1kbps. As for the greedy algorithm, which requires as much feedback as HYGIENE, its throughput and packet delay performance are quite poor. As far as fairness is concerned, results are so poor that no scheduler is really better than the others.

When reducing the scheduling period, in a LTE kind of way, to only a 1ms TTI instead of a 5ms frame, results are quite different. Reference algorithms seem to be better both on throughput and delay aspects. But these performances have a cost. While our algorithm only requires 5kbps of feedback per user, whatever the amount of available PRBs is, the reference scheduler needs 250kbps per user to handle 50 PRBs, which corresponds to the queue rate of 5 RT sessions, and even needs 500kbps to handle 100 PRBs.

If we compare both standards, with the same bandwidth, better performances are obtained with LTE, whose transmit power is 32W vs 20W for WiMAX. But feedback requirements are much more important for LTE. Because its scheduling period is 5 times shorter, it requires 5 times more feedback for the same amount of frequency bands. And because PRBs are roughly 2/3 times narrower than subchannels, more PRBs than subchannels are required to cover the same bandwidth, and LTE feedback load is even more increased.

To conclude, since the algorithm we designed only considers averaged wideband CQI values, its feedback load requirements are seriously reduced, but it can not benefit from frequency diversity. Channel opportunistic reference algorithms we studied, as for them, take advantage of frequency diversity, but waste some of their radio resources since they allocate subcarriers for whole frame/TTI durations. In WiMAX environment, with a long scheduling period, those reference wastes can be so important, especially at low queue rates, that it can not be compensated by frequency diversity gain. Moreover, frequency diversity has a huge feedback cost, which may not be affordable. However, when the

scheduling period is shortened, reference resource wastes do not penalize reference algorithms anymore, compared to the frequency diversity gain they benefit from, and as a result our wideband CQI strategy is no longer the best solution. Nevertheless, though the largest throughputs are provided by channel opportunistic schemes, they still require much more feedback than ours.

The DL results, both in WiMAX and LTE, have partially been presented at the 9th Management Committee Meeting of COST2100 at Vienna (Austria) in September 2009 [75].

Chapter 8

A GoB MIMO evolution of our heuristic

One of the main challenges in cellular networks evolution is channel capacity improvement. This chapter considers MIMO transmissions to improve frequency reuse, taking advantage of spatial orthogonality between users.

As we already mentioned in Section 4.1, there are several multiple antennas usages. They can improve reliability and combat fading through diversity, improve channel capacity through multiplexing, or improve frequency reuse through beamforming. As in CDMA, the technique which seems to be the most challenging from a scheduling point of view is the beamforming one.

To tackle the interference issue, users can be gathered into spatially compatible groups, where each user of a group is sufficiently distant to other members of its group so that all can use the same resources with only limited interferences. Multiple grouping techniques are compared in [76]. In opposition, users can also be gathered such as groups are independent from each other. So users of a group have to share the same resources, which can be reused from groups to groups. This is, for example, the case of the GoB strategy [44, pp. 51-55] that we already considered in Section 4.4, and which we are also going to use within this OFDMA framework.

8.1 A GoB evolution of our scheduler

As in CDMA, with the help of 4 transmit antennas, we can cover a 120° sector of a 3-sector cell with two series of 4 beams. But since OFDMA is based on orthogonal frequency bands, we can actually share the total

bandwidth between the two series of beams, and transmit simultaneously with all 8 beams on half the total bandwidth. We can, as a result, still benefit from a frequency reuse factor of 4. Indeed, different OFDMA subcarriers can receive different phase shifts, to be steered into different directions, whereas all simultaneous CDMA codes, conveyed through the same frequency band, can only be steered into the same direction.

As far as scheduling and resource allocation are concerned, as in CDMA, after a first step gathering users into space homogeneous groups, according to the beam index they fed back, we can simply run the SISO algorithms we already studied. In this case, we can reuse the 3 steps of Algorithm 6, as Algorithm 7 details.

The complexity of Algorithm 7 depends on each of its steps. Since STEP #1 considers all users one after another, the computational complexity is $\mathcal{O}(N_{users})$. Complexity of STEP #2 sorting stage is still of $\mathcal{O}\left(J_{BE} \cdot log(J_{BE})\right)$, where J_{BE} is still the amount of the largest BE session kind, typically the Bronze one, but only within one beam, and not for the whole sector. STEPS #3 and #4 are still of $\mathcal{O}\left(N_{symb} \cdot N_{subch}\right)$ complexity.

As far as feedback is concerned, added to channel quality and queue state informations already required in SISO, users have to feed back their best beam index. Since beam steering weights are pre-computed, users do not have to feedback accurate positions. For a fixed grid of 8 beams, 3 bits are sufficient for Channel Direction Indication (CDI). Added to the single 5-bit CQI value we already considered in SISO, each user has to feed its base station back with only 8bits per scheduling period, which correspond to 8kbps of feedback per user in LTE scenarios, or even 1.6kbps in WiMAX ones. There is no quantization issue since the feedback is limited to a beam index and a MCS level.

8.2 A MU-MIMO reference algorithm

As a reference scheduler, we implemented the iterative MIMO greedy algorithm proposed by [42] that we already considered in SISO scenarios and presented in Section 7.5.2. Exploiting near orthogonal conditions between beams, it tries to maximize the channel capacity, taking into account the channel response of each subchannel/PRB. Each user therefore has to feed back CQI for each subchannel/PRB plus CDI. The authors considered $4 \cdot N_{subch}$ bits of feedback per user per scheduling period, which is much more than our method. To handle 10 to 100 subchannels, this MIMO reference algorithm requires from 8kbps to 80kbps of feedback per user. It is even worse with the shorter scheduling period

Algorithm 7 GoB treemap-based 4-step scheduling algorithm.

STEP #1:
Gather users into groups of the same beam index;

STEP #2:
For each beam, sort sessions according to the hierarchy described in Section 7.3.1

$$\left\{U_{[1]},\ U_{[2]},\ \ldots, U_{[N_{GBR}]},\ \ldots, U_{[N_{GBR}+N_{RT}]},\ \ldots, U_{[N_{GBR}+N_{RT}+N_{BE}]}\right\};$$

STEP #3: for each beam, do:
$N_{available\ slots} \leftarrow \cdot N_{symb} \cdot \frac{N_{subch}}{2}$;
$k \leftarrow 1$; {current user for the loop}

while $N_{available\ slots} > 0$ **do**
Evaluate the capacity CS_k of a slot for the user U_k, according to TABLE 7.1 ;
Derive its required number of slots NT_k:

$$NT_k = \min\left\{\frac{Q_k}{CS_k},\ N_{available\ slots}\right\};$$

$N_{available\ slots} \leftarrow N_{available\ slots} - NT_k$;

if $k = N_{GBR} + N_{RT}$ **then**
$N_{available\ slots} \leftarrow N_{available\ slots} - N_{subch} + \mod\left(\sum_{u=1}^{k} NT_u,\ N_{subch}\right)$;
end if

$k \leftarrow k + 1$;
end while

STEP #4: Treemap resource allocation, for each beam:
Divide QoS vs non-QoS traffic after the $\left\lceil \frac{\sum_{u=1}^{N_{GBR}+N_{RT}} NT_u}{N_{subch}} \right\rceil$ symbol ;
Allocate resources to GBR traffic per row in the lower left corner area;
Allocate resources to RT traffic per row in the upper left corner area;
Allocate resources to BE traffic per row in the right hand side area;

of LTE, where the reference algorithm requires from 40kbps to 400kbps to handle 10 to 100 PRBs. In comparison, our algorithm, although not benefitting from channel diversity, only requires 1.6kbps for WiMAX scenarios, or 8kbps for LTE ones.

Since this is the MIMO version of the algorithm, several users can use the same subchannel/PRB, provided interference remains acceptable. As in SISO, the algorithm greedily looks for the best set of streams $\mathcal{P}^*$ that maximizes the weighted cell throughput $\mathcal{R}$:

$$\mathcal{R}(\mathcal{P}) = \sum_{(k,n)\in\mathcal{P}} w_k \cdot B_k(n) \tag{8.1}$$

To avoid recomputing throughputs each time a new stream is added to $\mathcal{P}$, the authors of [42] proposed a projection-based greedy (PBG) algorithm, where new streams are selected to be almost orthogonal to already selected ones. For our implementation, we considered near orthogonality where users are separated from more than a half-power half beamwidth (15° in our 4×4 MIMO scenarios). We also imposed a maximum of N_{Tx} beams per subchannel/PRB.

As far as the weights are concerned, as in SISO we chose

$$w_k = \begin{cases} 4 & \text{if } U_k \text{ is a GBR session} \\ 2 & \text{if } U_k \text{ is a RT session} \\ 1 & \text{if } U_k \text{ is a BE session} \end{cases} \tag{8.2}$$

to provide some priority to QoS-guaranteed sessions. Like in [42], uniformly distributed power has been considered. To handle interference, we suppose all beams active when evaluating throughputs, so we are sure adding a new stream does not increase interference and therefore does not reduce already estimated throughputs. Computational complexity of this algorithm is $\mathcal{O}(N_{subch} \cdot N_{queued} \cdot N_{Tx}^3)$, which is quite higher than ours.

8.3 DL WiMAX evaluation

Our first simulations were performed in DL WiMAX scenarios, with Advanced Antenna System (AAS) option, to enable the use of multiple antennas [70]. Simulation parameters are the same as our SISO computations, and are summarized in Table 8.1 We homogeneously distributed sessions within a 120° sector cell, according to distributions detailed in Appendix B, with the categories repartition of Table 7.2. We compare outage throughputs, cell throughputs, and packet delays of our GoB scheduler with the MU-MIMO reference algorithm of [42] and with a

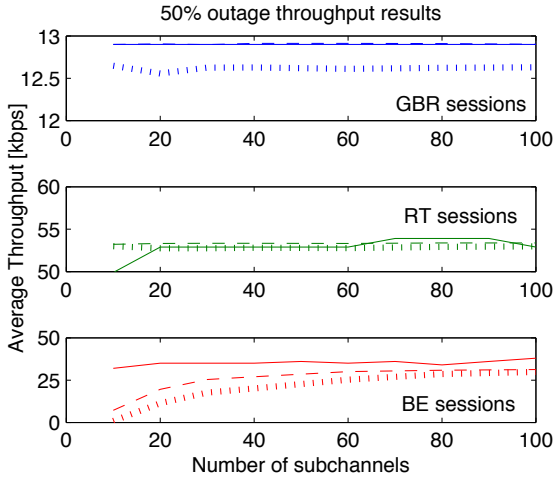
GoB extension of the bound we designed in Section 7.8. Results have been averaged out on 100 runs of 100 frames each, or due to computational time requirements on only 25 runs of 100 frames for the bound. As in SISO, we introduced intercell interference through G-factor.

Parameter	Value
Number of users	140 GBR + 120 RT + 180 BE
Traffic model	GBR: VoIP RT: Video streaming BE: HTTP
Channel model	Temporally correlated Rayleigh
Bandwidth	10 - 100 subchannels
Carrier Frequency	2GHz
Resource to share	Beams shared through Channel slots
User location	Uniformly distributed within a 120° sector cell (2.8km between BS)
User velocity	30kmph
Transmit power	20W
Noise power	-101.2 dBm
Interference model	No interference
Antenna pattern	4×4 MU-MIMO
TTI duration	5ms
Run duration	100 TTIs
Number of runs	100 runs (25 runs for bound evaluations)

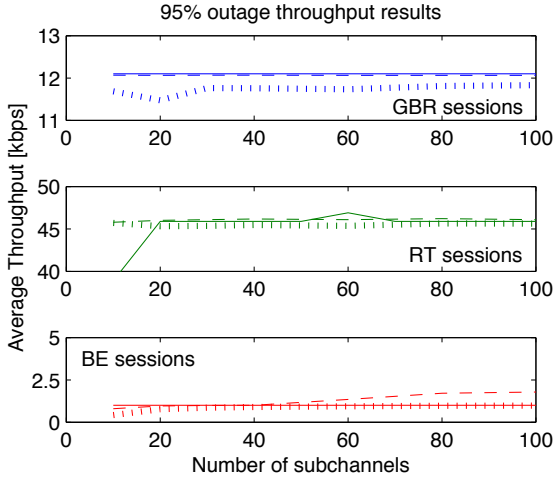
Table 8.1: Parameters for MU-MIMO DL WiMAX computations.

Throughput results are shown by Fig. 8.1, for 50% and 95% outage probability. As one can see, both algorithms can serve GBR and RT traffic, even with only 10 subchannels, whereas the bound undergoes some unfeasibility issues with only 10 subchannels. As far as BE traffic is concerned, throughputs remain way below channel requests. Our algorithm provides a quite better outage BE throughput, and at large bandwidth even outperforms the bound at 95% outage probability.

Per category cell throughput results are presented by Fig. 8.2. We retrieve that GBR and RT sessions are fully served, while BE throughputs remain way below demand. Our algorithm is able to provide better BE throughputs, but only when bandwidth is limited. Indeed, since our algorithm allocates frequency bands for as many symbols as required, it better uses bandwidth than the reference scheduler, which allocates



(a) 50% outage probability



(b) 95% outage probability

Figure 8.1: DL GoB WiMAX outage throughputs. Dashed lines stand for our scheme, dotted ones for the reference algorithm, and solid ones for the bound.

frequency bands for whole frame durations. When more bandwidth is available, it is the reference scheduler which better serves BE traffic, even matching the bound throughput, because its wastes are not an issue anymore, and because it benefits from channel diversity.

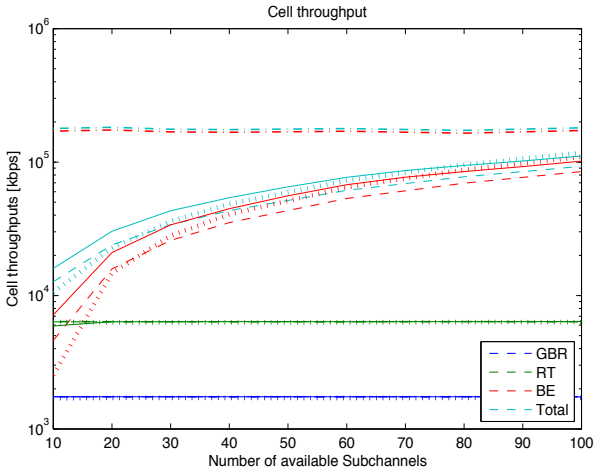
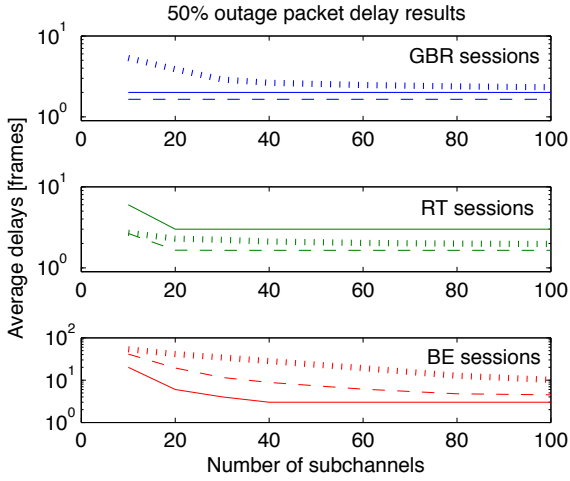


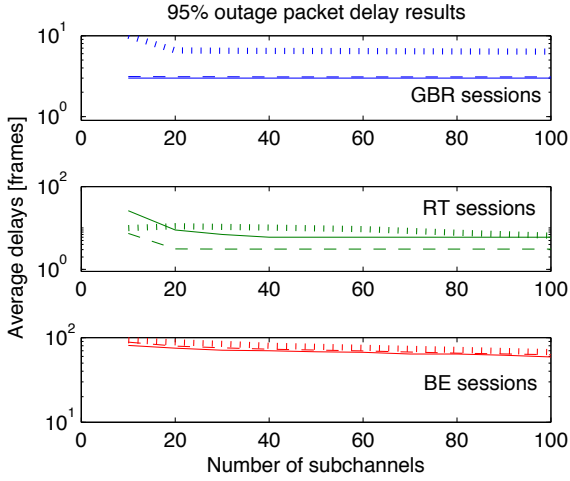
Figure 8.2: Cell throughput, in DL WiMAX. Dashed lines stand for our algorithm, dotted ones for the reference algorithm, solid ones for the bound and dash-dotted for the queue rates.

Let us point out that BE results of Figs. 8.1 and 8.2 are not contradictory. When there is enough bandwidth, algorithms provide the same high throughput to a few users. Same total throughputs can therefore be obtained while outage results are different.

We also analyzed outage packet delays, whose results are presented by Fig. 8.3. At 50% outage probability, our scheduler is able to deliver GBR packets within one frame duration, even when there are only 10 subchannels, and even beating the bound from a fraction of a frame. And from 20 subchannels on, RT packets are delivered by our scheduler within one frame duration too. In comparison, reference GBR and RT outage packet delays can not get shorter than respectively 5 and 10 frames. BE packets have to wait much longer, and some of them are actually never transmitted. The proportion of BE packets that are never transmitted can be estimated by the difference between BE throughputs and queue rates curves of Fig. 8.2. At 50% outage probability, our algorithm provides slightly shorter BE delays, but they remain poor in these overloaded scenarios. The longer delays for the reference algorithm can be explained by its less strict prioritization, and by its smaller channel



(a) 50% outage probability



(b) 95% outage probability

Figure 8.3: DL GoB WiMAX packet delay results. Dashed lines stand for our scheme, dotted ones for the reference algorithm and solid ones for the bound.

capacity when bandwidth is limited. As far as the bound delays are concerned, though limited to one frame at 50% outage or to two frames at 95% outage for GBR packets, they seem poorer for RT packet than other schedulers, though remaining lower than the real time constraint of a maximal 10-frame delay. This is probably what explains its much better performance on BE packet delays, especially at 50% outage.

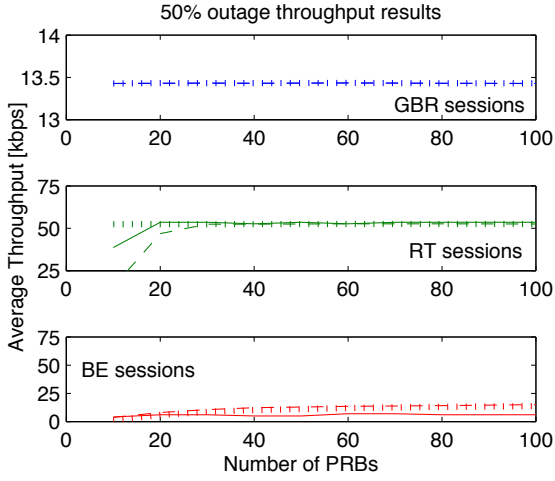
8.4 DL LTE evaluation

As in SISO, we also compared the scheduling algorithms in DL kind of LTE scenarios, with reduced scheduling period. We considered the LTE's transmission mode 5 [77], enabling Multi-User MIMO. We still used the same parameters as our SISO LTE simulations, summarized in Table 8.2, with the user distribution of Table 7.2, and still considered intercell interference through G-factor.

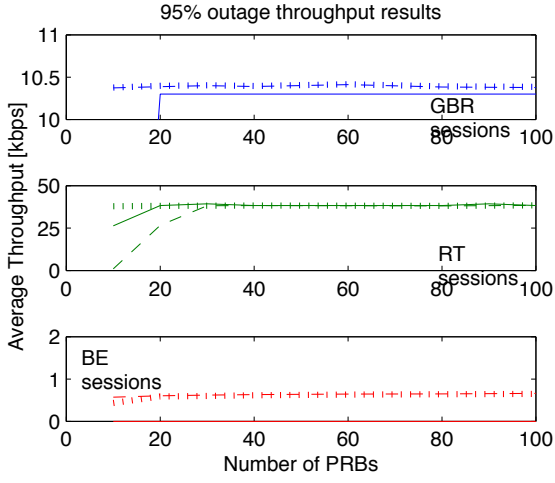
Parameter	Value
Number of users	140 GBR + 120 RT + 180 BE
Traffic model	GBR: VoIP RT: Video streaming BE: HTTP
Channel model	Temporally correlated Rayleigh
Bandwidth	10 - 100 subchannels
Carrier Frequency	2GHz
Resource to share	Beams shared through Channel slots
User location	Uniformly distributed within a 120° sector cell (3km between BS)
User velocity	30kmph
Transmit power	32W
Noise power	-101.2 dBm
Interference model	No interference
Antenna pattern	4×4 MU-MIMO
TTI duration	1ms
Run duration	100 TTIs
Number of runs	100 runs (25 runs for bound evaluations)

Table 8.2: Parameters for MU-MIMO DL LTE computations.

Outage throughputs are presented by Fig. 8.4. Both algorithms are



(a) 50% outage probability



(b) 95% outage probability

Figure 8.4: DL GoB LTE outage throughputs. Dashed lines stand for our scheme, dotted ones for the reference algorithm, and solid ones for the queue rates.

able to fully serve GBR and RT traffic and to match the bound when there are more than 20-30 PRBs. Our algorithm produces lower RT throughputs than the reference scheduler when bandwidth is really restricted. This is because the reference algorithm benefits from channel diversity and also because the LTE scheduling period is reduced compared to the WiMAX one. Indeed, the reference wastes, because PRBs are allocated for whole scheduling periods, are not an issue here. BE outage results remain poor for both algorithms. As in WiMAX, the bound also encounters unfeasibility issues with only 10 PRBs, which explains why the reference algorithm is able to beat it when there is such a limited bandwidth.

Cell throughput results are shown at Fig. 8.5. We still have the same performance for GBR and RT algorithms when bandwidth is sufficient, but the total BE throughput appears here slightly better with the reference algorithm, nearly matching the bound results. This is a consequence of the loss of frequency diversity of our algorithm. But let us remember that this loss is more than compensated by the feedback load reduction.

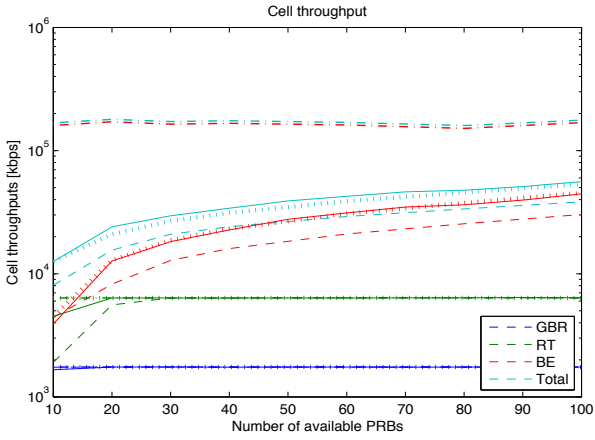
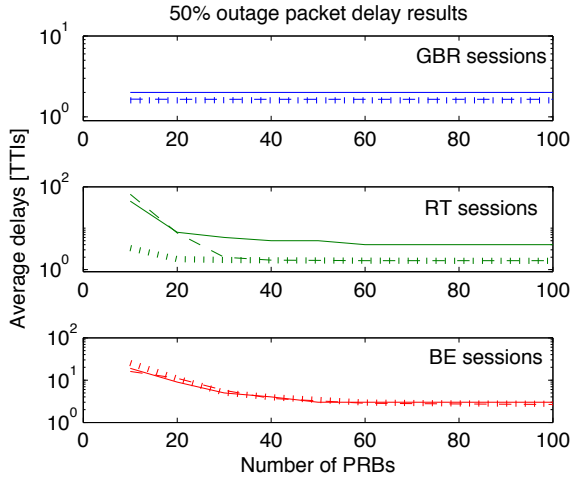
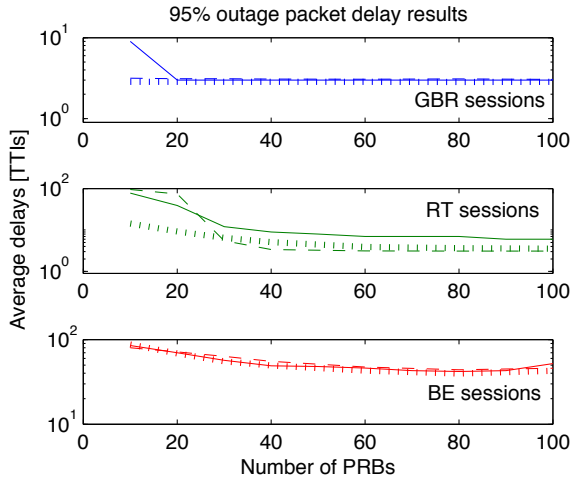


Figure 8.5: Cell throughput, in LTE scenario. Dashed lines stand for our algorithm, dotted ones for the reference algorithm, solid ones for the bound and dash-dotted ones for the queue rates.

Packet delay results are presented at Fig. 8.6. While both algorithms produce the same delay results for GBR and BE packets, the reference algorithm produces lower delays for RT packets when there are less than 30 PRBs. Let us recall that below 30 PRBs, our algorithm does not manage to fully serve the RT traffic. As far as the bound delays are concerned, they match those of our scheme as well as the reference one for GBR and BE packets, while it produces larger delays for RT packets.



(a) 50% outage probability



(b) 95% outage probability

Figure 8.6: DL GoB LTE packet delay results. Dashed lines stand for our scheme, dotted ones for the reference algorithm and solid ones for the bound..

When the bandwidth is sufficient, these RT delays however remains lower than the 10 TTIs real time constraint that the bound considers.

8.5 Concluding remarks

This chapter presented some MIMO evolution of our treemap-based algorithm presented in Chapter 7 through a GoB strategy. This is a quite simple algorithm, with per user feedback limited to 8kbps in LTE, or even to 1.6kbps in WiMAX, whatever the available bandwidth is. Based on beam orthogonality, it provides a frequency reuse factor of N_{Tx} per 120° sector cell.

The algorithm first gathers users according to their location in the sector, to match one fixed beam of the GoB. Then for each beam, it sorts out users according to the hierarchy described in Section 7.3.1, evaluates their resource needs expressed in slots, and uses a treemap-based scheme to allocate these slots.

Scripted in MATLAB, we compare our scheduler to an iterative greedy algorithm, which takes into account the channel response of each subchannel/PRB, therefore having much higher feedback requirements. We also compare it to a GoB extension of the bound we designed in Section 7.8. In WiMAX, our algorithm provides higher throughputs and lower delays than the reference scheme. In LTE, due to its shorter scheduling period, outage throughputs and delays of both algorithms are comparable, and the reference scheme is able to match the cell throughput of the bound.

As a result, like in SISO, the loss of frequency diversity for our scheme is more than compensated by its reduction of wasted slots, compared to the reference algorithm in WiMAX scenarios with long scheduling periods. But when considering shorter scheduling periods, such as the 1ms TTI of LTE, reference wastes are not an issue anymore, and our algorithm, not benefiting from frequency diversity, though able to provide nearly the same outage results when bandwidth is sufficient, can not allow the same cell throughput as the opportunistic channel reference scheduler we considered. With both standards, our algorithm manages to reduce tremendously the feedback load. As far as the bound is concerned, benefiting from frequency diversity and avoiding the wastes of the reference algorithm, when bandwidth is sufficient, it is able to provide, for both WiMAX and LTE the best throughputs and the lowest GBR and BE delays. RT delays are a slightly larger but remain under the 10 scheduling periods boundary.

Chapter 9

WiMAX Burst mapping

Up until now, we have considered that we can allocate slots of resource as we want. Actually, there are some constraints. This chapter focuses on the particular constraints of the WiMAX standard.

The WiMAX resource allocation is performed by slots, which represent the usage of one subchannel during one symbol. The MCS level and the other parameters determining the throughput a user gets from one slot are gathered into a PHY-profile. The allocation is done per burst, where a burst contains the datas of one or several users. Only a single PHY-mode can be used in a burst.

9.1 Mapping constraints

A frame is divided into two parts, the DL and the UL subframes, which can use different frequency bands (FDD), or be time alternated (TDD). The burst mapping problem is quite different in DL and UL subframes. Figure 9.1 presents an example of the resource allocation of a WiMAX TDD frame. Frames are separated from each other by a Receive/Transmit transition gap (RTG), and a Transmit/Receive transition gap (TTG) separates the DL and the UL subframes of a frame.

The DL subframe is first composed of a one symbol duration preamble, followed by the Frame Control Header (FCH), and the DL- and UL-MAPs, which specify the PHY-profile of each burst allocation. Then there are the burst allocations. Each burst must have a rectangle allocation, due to the format of the allocation MAP, which specifies its shape and position through its Information Elements (IE). So the more bursts there are, the bigger the DL-MAP will be. Gathering users into a minimal number of bursts is therefore preferable, to limit the signaling burden. Within a burst, the unique connection identifier (CID) separates each user's transmissions.

In UL, resource allocation is different. Bursts are mapped as continuous slots in a row wise order. So the UL mapping problem is straightforward. It is actually pretty close from what we have done within Algorithms 6 and 7.

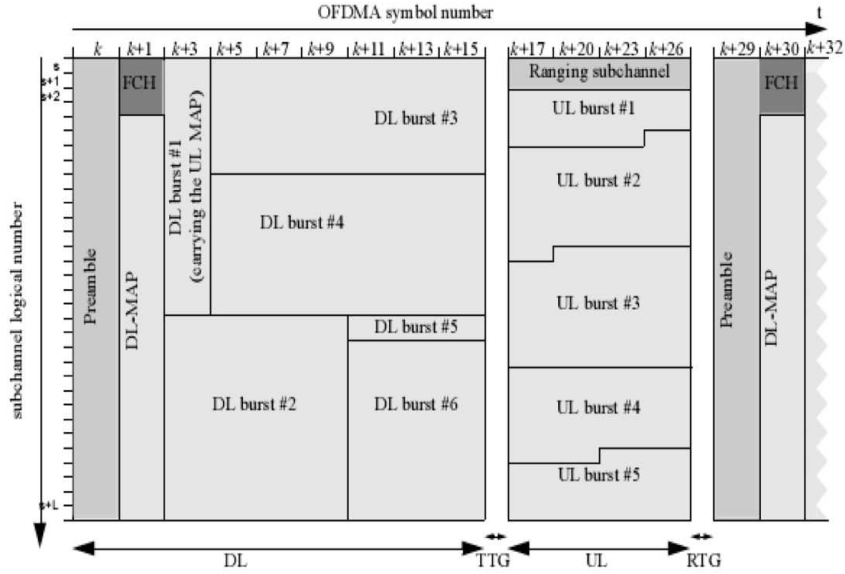


Figure 9.1: WiMAX TDD frame structure, taken from [3].

As far as DL is concerned, this tiling problem is actually a bi-dimensional bin packing problem. It is a variation of the NP-HARD knapsack problem [78]. Only a few heuristics have been applied to this problem, and generally waste a lot of slots to match the rectangle constraint on the allocations shape. The authors of [79] presented a heuristic rounding burst requirements of slots to a multiple of the number of subchannels, which is quite simple but leads to an important waste of slots. Other strategies like SDRA [71] or the Raster-based algorithm [80] waste no slots in rounding operations, but at the price of multiplying the number of bursts, therefore wasting slots in DL-MAP overhead. The authors of [55, 56] came up with some variations of what they called the Sequential Rectangle Placement (SRP) problem. They separate bursts into 3 categories, according to their size, and map them into vertical stripes of growing width. A different stripe gathering strategy is introduced by the OCSA algorithm [81, 82]. This technique is more complex, but leads to fewer wasted slots.

9.2 Our burst mapping algorithm

Once tuned to meet WiMAX requirements, the Squared treemap visualization algorithm [68] described in Section 7.1 seemed to us quite interesting to solve the DL burst mapping problem. The application is however not straightforward. Different users can not share the same subchannel during the same symbol time. Ceiling operations have to be introduced, both on time dimension at symbol scale and on frequency dimension at subchannel scale. The amount of wasted slots due to ceiling operations has to be limited. The allocation criteria we are going to consider is therefore the amount of wasted slots, which we try to minimize, instead of the aspect ratio considered by the original visualization algorithm of [68]. We named that heuristic *sqTM*, since it is based on the Square treemap algorithm.

Given the slots requirements of each scheduled burst, the objectif of sqTM is hence to produce a resource allocation mapping where each burst receives a rectangular shaped allocation while minimizing the total amount of wasted slots.

9.2.1 The allocation procedure

Bursts are treated sequentially, with no requirement for them to be sorted out. The allocation is done through vertical stripes of growing width. At first, a stripe is only composed of one burst, with full height, and ceiled width. The number of wasted slots is evaluated.

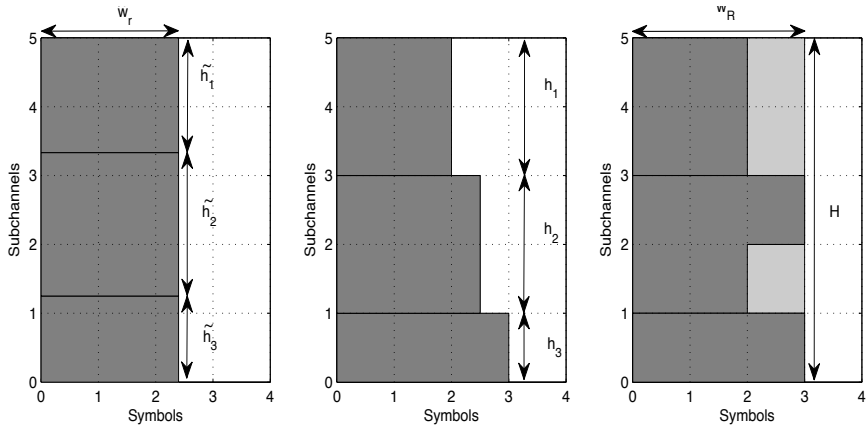
Then we sequentially try to add each of the other bursts to that stripe. The number of slots wasted by the stripe is evaluated taking into account all ceiling losses we describe in the following Section 9.2.2. Each added burst which reduces the number of wasted slots is effectively added to the stripe.

When adding any of the remaining bursts can not further reduce the amount of slots wasted within the current stripe, its content is finalized. Another stripe is started, where all remaining bursts are tried to be added, and so on.

The first stripe is built to include the FCH and the DL-MAP.

9.2.2 Ceiling operation

Because burst mapping divisions can only happen between subchannels and between symbols, and can not split them, some rounding operations have to be done. Let us consider here a stripe R containing bursts b_i requiring r_i slots, with $i = 1, \ldots, n_B$. The height of the frame is the number of available subchannels N_{subch}.



(a) Accurate allocation (b) Rounded heights (c) Ceilled allocation

Figure 9.2: Ceiling operation, with bursts of size 4, 5 and 3 slots. After ceiling, 3 slots are wasted.

A first allocation is performed, with no ceiling, assuming we can split subchannels and symbols, as Fig. 9.2(a) shows. The temporary width $\tilde{w}_R$ of the stripe is derived from:

$$\tilde{w}_R = \frac{\sum_{b_i \in R} r_i}{N_{subch}} \tag{9.1}$$

and the temporary height $\tilde{h}_i$ of each burst is given by

$$\tilde{h}_i = \frac{r_i}{\tilde{w}_R} \quad \forall i = 1, \cdots, n_B. \tag{9.2}$$

Then the height allocated to each burst, which has to be an integer number of subchannels, is rounded to h_i, paying attention that the sum of all rounded heights must match the total bandwidth. Defining the operator $\lceil \cdot \rfloor$ such as it rounds to the closest integer, we have:

$$h_i = \begin{cases} \left\lceil \tilde{h}_i \right\rfloor & \forall i = 1, ..., n_B - 1; \\ N_{subch} - \sum_{j=1}^{n_B - 1} h_j & i = n_B. \end{cases} \tag{9.3}$$

As shown on Fig. 9.2(b), according to the rounded heights, the width of each burst has to be adapted to $\tilde{\tilde{w}}_i$:

$$\tilde{\tilde{w}}_i = \frac{r_i}{h_i} \quad \forall i = 1, \cdots, n_B. \tag{9.4}$$

Finally, the width of the stripe w_R, which is also the final width w_i of each burst, can be fixed to:

$$w_R = \max_{i=1,\cdots,n_B} \left\lceil \tilde{\tilde{w}}_i \right\rfloor. \tag{9.5}$$

where the operator $\lceil \cdot \rceil$ is defined such as it rounds to the closest larger integer.

The number t_{wasted} of slots wasted in the stripe allocation is obtained from

$$t_{wasted} = (N_{subch} \cdot w_R) - \sum_{i=1}^{n_B} r_i. \tag{9.6}$$

On the example of Fig. 9.2(c), it corresponds to the light grey slots.

These evaluations represent an amount of computations which can sometimes be avoided. The ceiling operation is only fruitful when the stripe composition provides a reduction of the amount of wasted slots. After the first allocation of Fig. 9.2(a), even if there would not be any waste due to rounded heights, we are sure we at least waste $N_{subch} \cdot (\lceil \tilde{w}_R \rceil - \tilde{w}_R)$ slots. Therefore if this minimum amount of wasted slots is larger than the waste of the previous stripe composition, we can be sure the current stripe composition is worse than the previous one. Other ceiling computations are therefore useless, and can be skipped.

9.2.3 The sqTM Algorithm

Our burst mapping sqTM procedure can therefore be summarized into Algorithm 8. While there remains unallocated bursts, it starts a new stripe R, and sequentially tries to add each of the remaining bursts b_i. If the stripe is empty, the burst is added to the stripe (lines 4 to 6). If not, each remaining burst is tested, and the ceiling operation of Section 9.2.2 is performed to evaluate the number of wasted slots.

The test of line 9 prevents the algorithm from performing computations of lines 10 to 12, plus the evaluation of line 13 when it knows, a priori, they are pointless.

Finally, after all computations, if the amount of wasted slots is reduced by adding the burst b_i to the current stripe (test of line 13), it is definitely added, and remaining bursts are then tried to be added as well.

9.2.4 Computational complexity

The algorithm is composed of a main *while* loop bounded by the total number N_{bursts} to be mapped. For each stripe, all unallocated bursts, which can be up to N_{bursts}, are tested. The computational complexity of sqTM is therefore $\mathcal{O}({N_{bursts}}^2)$.

Algorithm 8 The sqTM algorithm

while some unallocated bursts remain **do**
  Start a new stripe R;
  for each unallocated burst b_i **do**
    if stripe R is empty **then**
      Add b_i into R;
      $t_{wasted} \leftarrow (N_{subch} - r_i \bmod N_{subch})$
    else {R is non empty}
      $\tilde{w}_R \leftarrow \sum_{b_j \in R \cup \{b_i\}} r_j / N_{subch}$;
      if $(\lceil \tilde{w}_R \rceil - \tilde{w}_R) \cdot N_{subch} < t_{wasted}$ **then**
        $h_j \leftarrow \lceil r_j / \tilde{w}_R \rfloor \quad \forall b_j \in R$;
        $h_i \leftarrow N_{subch} - \sum_{b_j \in R} h_j$;
        $w_R \leftarrow \max_{b_j \in R \cup \{b_i\}} \lceil r_j / w_R \rceil$;
        if $(N_{subch} \cdot w_R) - \sum_{b_j \in R \cup \{b_i\}} r_j \leq t_{wasted}$ **then**
          Add b_i into R;
          $t_{wasted} \leftarrow (N_{subch} \cdot w_R) - \sum_{b_j \in R \cup \{b_i\}} r_j$;
        end if
      end if
    end if
  end for
end while

9.3 Some reference burst mapping algorithms

To assess the performance of our sqTM burst mapping algorithm, we compare it to four reference procedures, namely the bucket-based algorithm [79], SDRA [71], SRP [55] and OCSA [81] we describe hereafter.

9.3.1 Bucket-based algorithm

The authors of [79] proposed a simple algorithm, where burst sizes are ceiled to a multiple of the number of subchannels. Then bursts sequentially receive as many one-symbol width consecutive columns as they require. The algorithm is very simple, of complexity $\mathcal{O}(N_{bursts})$, but wastes a lot of slots. Indeed, in average $\frac{N_{subch}-1}{2}$ slots are wasted per burst, since the last column of a burst can equally contain 1 to N_{subch} slots, therefore wasting $N_{subch} - 1$ to 0 slots.

9.3.2 SDRA

The SDRA algorithm of [71] has also a column filling strategy, but does not ceil bursts size. When the allocation of one burst is done, the algorithm continues filling the same one-symbol width column with the allocation of the following burst. As a result, no slot is wasted on ceiling operations, and the algorithm is of complexity $\mathcal{O}(N_{bursts})$.

To meet the rectangular shape requirement, each burst actually receives multiple rectangles, up to 3 ones: a first rectangle to complete the current one-symbol width column, another rectangle of several full columns, and another one to complete its allocation on a new column. So the DL-MAP, whose size depends on the number of rectangle allocations, is increased, and some slots are wasted in signaling overhead.

9.3.3 SRP

The authors of [55] modeled the burst mapping problem as a Sequential Rectangle Placement. They defined 3 kinds of bursts, according to their size:

- the small bursts, with $r_i \leq 2\sqrt{N_{subch}}$. These ones can only have one-symbol width allocation.
- the medium bursts, with $2\sqrt{N_{subch}} < r_i \leq \frac{N_{symb}}{2}\sqrt{N_{subch}}$, receiving maximum $\frac{N_{symb}}{2}$-symbol width allocation.
- the large bursts, with $r_i > \frac{N_{symb}}{2}\sqrt{N_{subch}}$, with maximum $2N_{symb}$-symbol width allocation.

where N_{symb} is the number of symbols of the downlink scheduling period, and categories threshold on bursts size are designed to reduce the amount of wasted slots.

The authors also defined the concept of job sets, which are column shaped and contain only bursts of the same kind. The maximum width of a job set JS, of size $|JS|$, is the maximum width of its bursts, and is denoted by MAX(JS). The width of a job set R is denoted WIDTH(JS) and is evaluated by:

$$\text{WIDTH}(JS) = \min \left\{ \left\lceil \frac{\sum_{b_j \in JS} r_j}{N_{subch} - |JS|} \right\rceil \ , \ \text{MAX}(JS) \right\} \tag{9.7}$$

As algorithm 9 details, SRP sequentially handles each burst, and tries to add it into the current job set of its kind (test of line 3). If there remains not enough room for it, the job set is closed, and another one is created (lines 8 to 10). SRP is therefore also of complexity $\mathcal{O}(N_{bursts})$, even if it requires more computations than the two other algorithms.

Algorithm 9 The SRP algorithm

```
1:  for i = 1, ..., n_B do
2:      Let JS be the job set of b_i ;
3:      if Σ_{b_j ∈ JS ∪ {b_i}} ⌈ r_j / MAX(JS) ⌉ ≤ H then
4:          JS ← JS ∪ {b_i};
5:          if r_i > 2√H then {medium or large burst}
6:              ROWS(JS) ← min { ⌈ Σ_{b_j ∈ JS} r_j / (H − |R|) ⌉ , MAX(JS) };
7:          end if
8:      else {No room in JS for b_i}
9:          Close JS ;
10:         Open a new job set for b_i with ⌈ r_i / H ⌉ rows;
11:     end if
12: end for
```

9.3.4 OCSA

The OCSA algorithm of [81, 82] first requires bursts to be sorted in a descending order (largest area first). The largest unallocated burst is placed first, in a vertical stripe as narrow as possible. OCSA then tries to place on the top of that burst other bursts fitting in the stripe without modifying its width. When no more bursts can be added on the top of it, another stripe is started with the largest unallocated burst remaining, and so on.

Since there can be N_{bursts} stripes, and for each one we try to put on the top of it all unallocated bursts, which can also be as many as N_{bursts}, the total complexity of OCSA is $\mathcal{O}(N_{bursts}^2)$.

9.4 Burst mapping performances

To evaluate the performance of our sqTM algorithm, and compare it with the four reference algorithms we presented in Section 9.3, we scripted all these algorithms in MATLAB. After detailing our simulation parameters, we show some snapshot allocation results, reflecting how algorithms work. We then present averaged performance results on wasted slots and compare algorithms complexity.

9.4.1 Simulation parameters

We focused our comparisons on TDD DL WiMAX with an DL:UL ratio of 2:1, so the DL subframe has a 29-symbol duration, with the first symbol dedicated to preamble. We considered 10MHz of bandwidth, divided into 30 subchannels. The burst mapping area is therefore of size 28×30, which corresponds to 840 slots.

As [83] mentions, the FCH represents 24 bits. The DL-MAP has a fixed 88-bit part, plus 60 extra bits per burst. The FCH and the DL-MAP are repeated 4 times and are transmitted using QSPK 1/2. Since we only consider DL, the UL-MAP is acted by a random size burst.

As far as PHY-modes are concerned, we consider the 9 MCS levels of Table 7.1, from QPSK 1/2 to 64QAM. Assuming users are gathered into bursts according to their PHY-mode to reduce DL-MAP size, algorithms have to map the FCH, the DL-MAP and 9 bursts plus an extra one acting as the UL-MAP. With the 0.937 bit/s/Hz spectral efficiency of QPSK 1/2 [83], FCH requires 4 slots, and a 10-burst DL-MAP requires 102 slots.

For these first simulations, aiming to compare burst mapping algorithms without scheduling influence, we generated bursts of random size such as the sum of all bursts plus the FCH and a 10-burst DL-MAP requirements correspond to the 840 slots. As a result, due to ceiling wastes or signaling overhead, some bursts do not fit within the subframe. We compare the amount of slots overflowing from the DL subframe as a performance indicator to compare mapping algorithms.

9.4.2 A sample output result

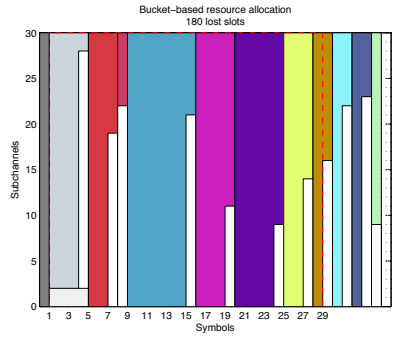
We present first a representative snapshot allocation performed by all algorithms. We randomly drew the bursts size of Table 9.1.

FCH	DL MAP	UL-MAP	Burst #1	Burst #2	Burst #3
4	102	71	8	189	109
Burst #4	Burst #5	Burst #6	Burst #7	Burst #8	Burst #9
141	76	44	38	37	21

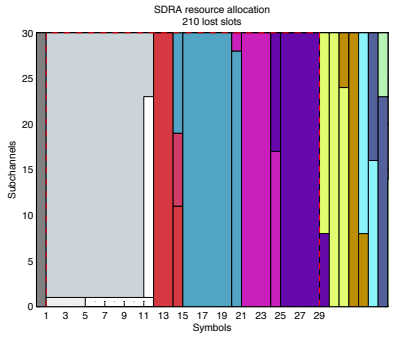
Table 9.1: Sample snapshot of burst sizes in slots.

The allocations obtained by the reference algorithms are presented by Fig. 9.3, while sqTM allocation is shown at Fig. 9.4. The first dark grey stripe is the preamble and the two light grey rectangles on the lower left hand corner are the FCH and the DL-MAP. White slots are unused.

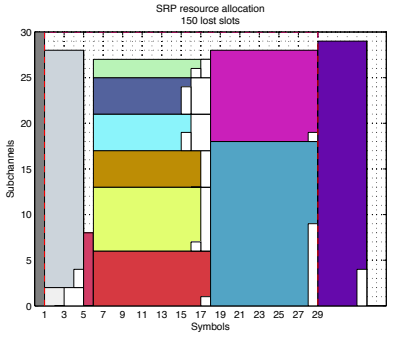
Reference algorithms waste a lot of slots, between 150 and 210, much more than our algorithm, which only wastes 30 slots. One can see on Fig. 9.3(a) that the strategy of the bucket-based algorithm leads to an average waste of $\frac{N_{subch}-1}{2}$ per burst, with all the white rectangles wasted in the lower right corner of each burst allocation. The SDRA algorithm, whose allocation is shown on Fig. 9.3(b), produces a DL-MAP of size more than doubled compared to the 3 other reference algorithms. So



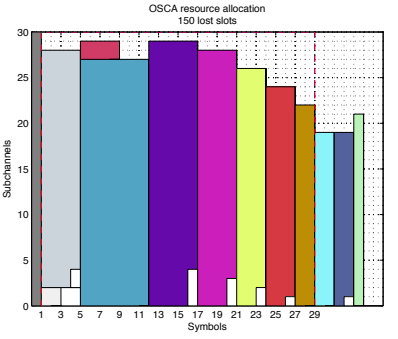
(a) Bucket-based



(b) SDRA



(c) SRP



(d) OCSA

Figure 9.3: Reference algorithms sample allocation

even if it does not waste slots on ceiling operations, in the end it wastes a lot of slots just the same. As Fig. 9.3(c) shows, SRP makes large stripes of multiple piled bursts, leading to the same wastes as OCSA and its thin stripes of Fig. 9.3(d).

As far as our sqTM algorithm is concerned, thanks to its growing stripes strategy, where decisions are taken to minimize wasted slots, it manages to limit wastes to a single overflowing column of 30 slots, as Fig. 9.4 shows.

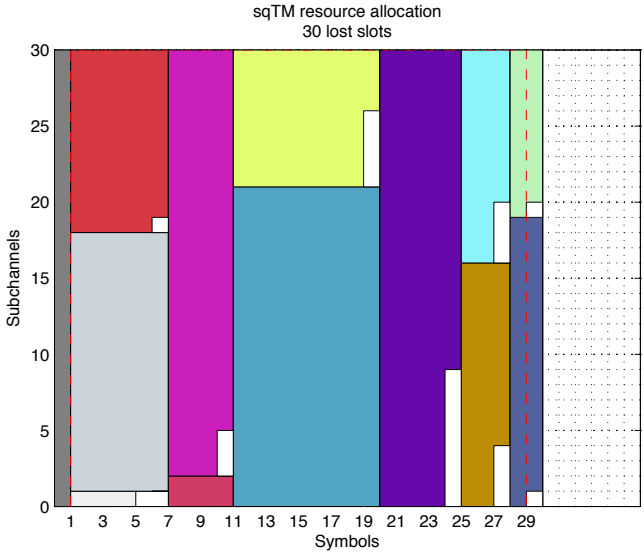


Figure 9.4: sqTM sample allocation

The authors of [81] also pointed out that resource mapping should be done aiming at minimizing users energy consumption. They therefore recommend that bursts duration should be minimized, to limit the time mobile users are listening to the base station. This is the reason why they designed OCSA to make thin rectangles of resource. Having a look at Figs. 9.3 and 9.4, it appears that thinness of allocations of the Bucket-based algorithm and SDRA is actually quite similar as thinness obtain with OCSA. Our sqTM algorithm produces slightly wider allocations than these three algorithms, but not as wide as SRP. Its performance on mobile user consumption are therefore only average.

9.4.3 Averaged results

For a more accurate evaluation of the algorithms, we also averaged performances over 100,000 allocations. The cumulative distributions and

the averaged value of the amount of wasted slots per allocation for each algorithm are presented by Fig. 9.5. Curves are not smooth since the amount of wasted slots is a multiple of the number of subchannels, which is 30 in our example.

As Fig. 9.5 shows, sqTM seriously reduces the amount of wasted slots, by 60% compared to OCSA, of even by 75% compared to SDRA. Considering the 840 slots of the simulation, in average sqTM only wastes 6% of them, while the bucket-based algorithm, SDRA, SRP and OCSA respectively waste 19%, 25%, 20%, and 15% of the available slots, which is quite a lot.

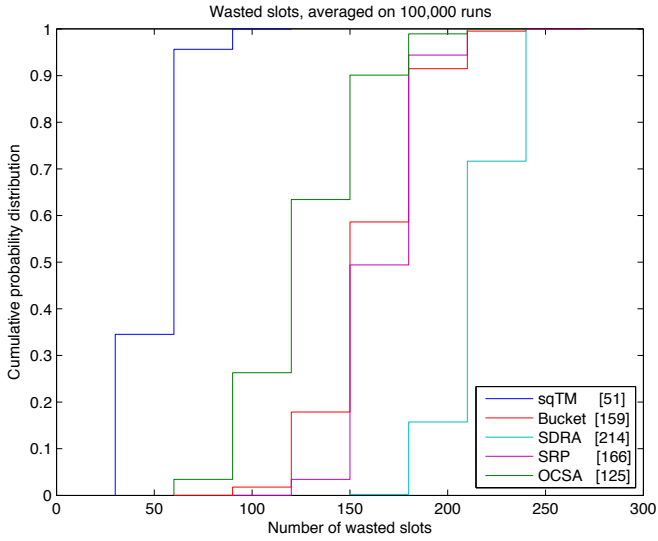


Figure 9.5: Per allocation wasted slots distributions.

9.4.4 Computational complexity

As we have already shown, sqTM is much more efficient than reference algorithms, but it is also more complex. Indeed, sqTM is a quadratic algorithm, while most of the reference algorithms are linear. But actually, when users are gathered into a minimal number of bursts, it is more the computational time which is important. For that purpose, Fig. 9.6 presents the sorted computational time algorithms required for each of the 100,000 allocations of our simulation, while Table 9.2 compares computational complexities to average and 95% worst time computational time. And as one can see, linear SDRA and SRP take more time than OCSA, which is quadratic.

Algorithm	Complexity	Mean time	95% worst time
sqTM	$\mathcal{O}(N_{bursts}^2)$	367 μs	457 μs
Bucket	$\mathcal{O}(N_{bursts})$	53 μs	54 μs
SDRA	$\mathcal{O}(N_{bursts})$	230 μs	241 μs
SRP	$\mathcal{O}(N_{bursts})$	284 μs	309 μs
OCSA	$\mathcal{O}(N_{bursts}^2)$	175 μs	180 μs

Table 9.2: Mean and 95% worst time computational times, averaged on 100,000 allocations.

As far as sqTM is concerned, it required on average 367 μs per allocation. It is therefore 2 times slower than OCSA, or even 6 times slower than the bucket-based allocation. But despite heavier computations, these mean 367 μs, or even the worst time 457 μs, though obtained on a laptop running MATLAB and not on a real base station device, are one order of magnitude shorter than the 5 ms frame duration. So complexity should not really be an issue.

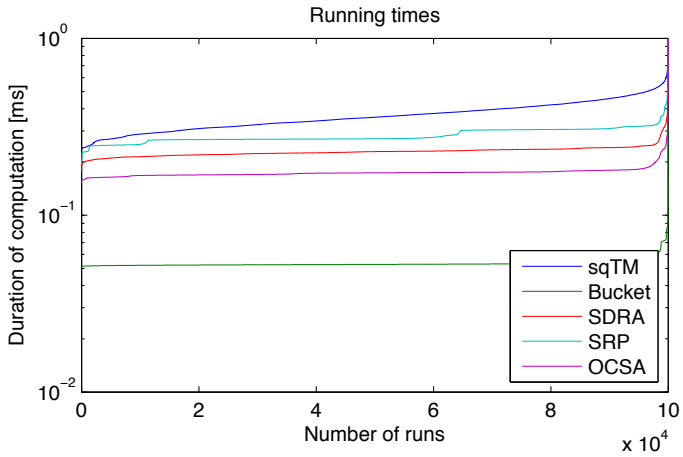


Figure 9.6: Computational time for burst mapping algorithms.

9.4.5 Concluding remarks

This first analysis of sqTM shows that our new burst mapping algorithm seems more efficient than the reference algorithms we compared it with. We obtained a reduction of wasted slots between 60% and 75%, at the price of an increased but still acceptable computational time. This waste reduction should provide better cell performances, which is what we are going to study in the next section.

Those first results have been accepted for presentation to the IEEE 74th Vehicular Technology Conference VTC2011-Fall, held in San Francisco (United States) in September 2011 [84].

9.5 Influence of mapping on cell performances

As we have shown in the previous section, the efficiency of burst mapping algorithms can significantly modify cell capacities. On pure mapping problems, sqTM showed seriously improved performance, and within this section, we study the influence of these improved performance on a full scheduling scheme. We compare outage throughputs and packet delays obtained with different burst mapping algorithms for the same scheduling algorithm.

9.5.1 Scheduling algorithm

As a scheduling algorithm, we consider the 3-step Algorithm 6 we introduced in Section 7.3. It offers QoS guarantees, has a low feedback requirement, and is rather simple. It first sorts out sessions according to a QoS-based hierarchy, then assesses sessions needs in slots, according to their channel quality, and selects users granted for transmission. When transmitting sessions are identified by the scheduler, before mapping, packets are gathered into bursts of homogeneous PHY-mode. Afterwards, the third step, which is the mapping one, is replaced by sqTM or by any of the reference burst mapping algorithms.

Because burst mapping wastes slots, some bursts may actually not be mapped within the current DL subframe, and the packets they contain, even GBR ones, have to be delayed. To limit this issue, the DL subframe is considered one symbol narrower when selecting users granted for transmission.

9.5.2 Simulation parameters

As in Section 9.4, we consider DL TDD WiMAX, with a DL:UL ratio of 2:1, and reuse the same parameters. The difference is that now, bursts size are derived from our scheduling algorithm, which runs on the same parameters as Section 7.9.

We therefore have 140 GBR sessions, 120 RT ones and 180 BE ones, whose packets are sorted and selected for transmission by Algorithm 6. Selected packets are then gathered into bursts according to their PHY-

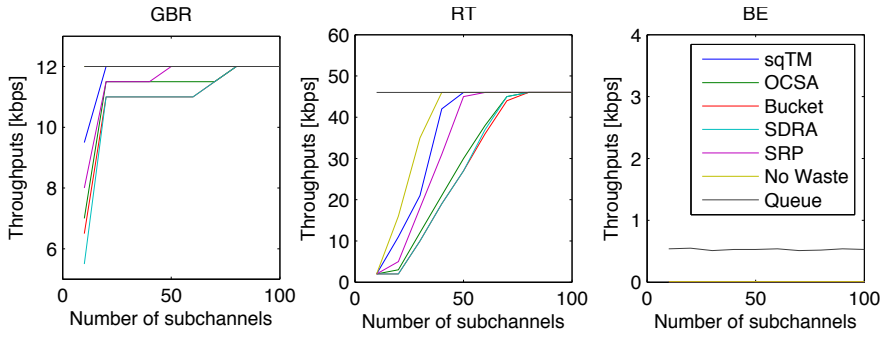
mode. Considering the 9 MCS levels of Table 7.1 as PHY-modes, we can have up to 9 bursts to map. However, we may have fewer bursts as well.

To limit the impact of scheduled bursts which can not be mapped, we handle them in descendending order of spectral efficiency. It is of course better to delay QPSK slots than 64QAM ones.

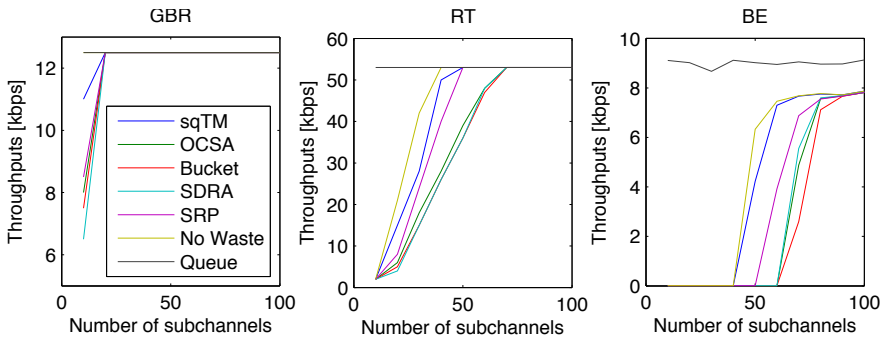
Let us also point out that since we focus only on the DL subframe, we do not consider any UL-MAP to map in the DL subframe.

9.5.3 Outage throughputs results

The throughputs we obtained at 95% and 50% outage probability for 10 to 100 subchannels are displayed by Fig. 9.7. For each kind of traffic, the throughput each burst mapping algorithm allows is compared to an ideal allocation with no ceiling wastes and to the queue rates. These results are averaged on 100 runs of 100 frames.



(a) 95% outage throughputs



(b) 50% outage throughputs

Figure 9.7: Burst mapping influenced outage throughputs.

As one can see on Fig. 9.7, the ideal mapping algorithm (No Waste) manages to always fully serve GBR traffic, and to fully serve RT traffic as well, whenever there are at least 40 subchannels. BE traffic results remain quite poor. When it comes to comparing burst mapping algorithms, sqTM clearly outperforms reference algorithms. It only requires 20 subchannels to fully serve GBR traffic, while SRP, the second best algorithm requires at least 50 subchannels for the same performance, and other algorithms at least 80 ones. RT and BE results also point out a significative throughput improvement obtained by sqTM.

Cell throughputs, presented by Fig. 9.8, also demonstrate that wasted slots reduction provided by sqTM allows significative throughput improvements. For example with 50 subchannels, sqTM allows double outage cell throughputs compared to OCSA, SDRA or the Bucket-based algorithm. And from 60 subchannels on, sqTM becomes insensitive to wasted slots, while reference algorithms require at least 70-80 subchannels for the same performances.

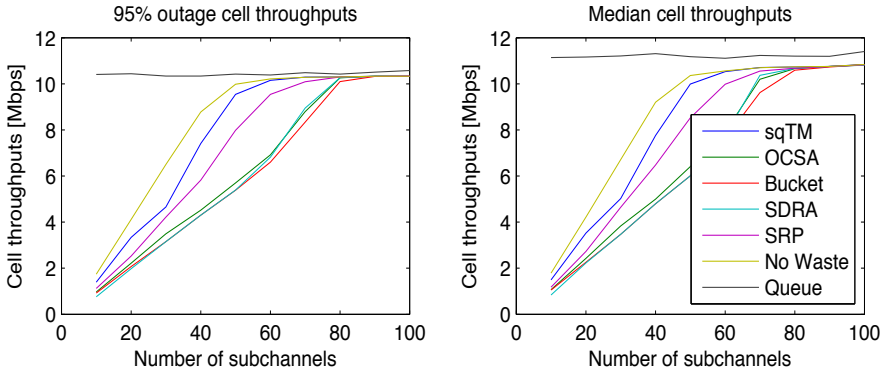
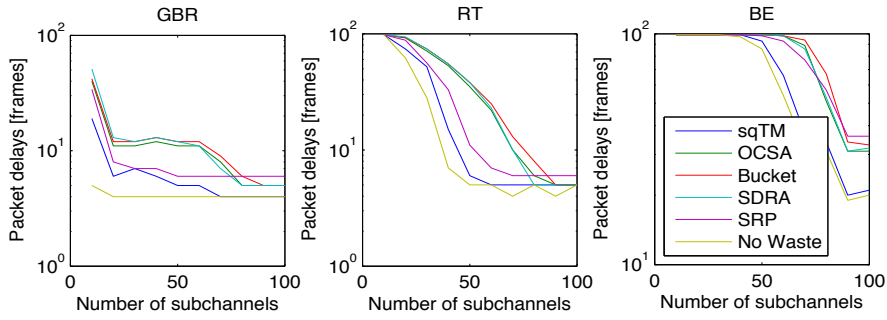


Figure 9.8: Burst mapping influenced cell throughputs

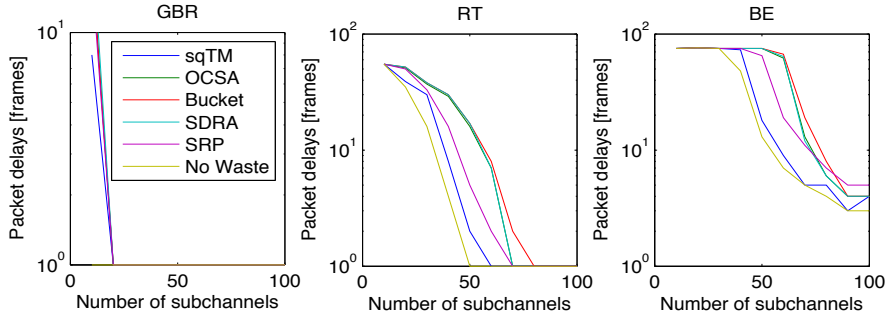
9.5.4 Outage packet delays results

To figure the delay increase due to scheduled bursts that can not be mapped, we also considered outage packet delay results as a burst mapping performance indicator. These results are presented by Fig. 9.9.

As Fig. 9.9 shows, sqTM delays are much lower than those obtained with the reference algorithms, and are actually not so far away from those obtained with the ideal mapping. SRP requires at least 10 more subchannels to reach the same delays as sqTM, while the other reference algorithms may require 20 to 30 more subchannels.



(a) 95% outage packet delays



(b) 50% outage packet delays

Figure 9.9: Burst mapping influenced outage packet delays.

9.5.5 Concluding remarks

As preliminary results of Section 9.4 suggested, the reduced amount of wasted slots by our sqTM burst mapping algorithm has a significative effect on both throughputs and packet delays. Compared to the burst mapping reference algorithms we implemented, sqTM requires 10 to 30 less subchannels to reach the same performances.

As a result, a practical system relying on sqTM would require less bandwidth to ensure the same throughput and packet delay guaranties than other systems relying on reference mapping algorithms, and from the same bandwidth, sqTM allows better throughput and packet delay performance than reference mapping algorithms.

9.6 Burst mapping conclusions

This chapter focused on radio resource allocation to scheduled users. Considering specifically DL WiMAX, burst mapping requires that allocations have rectangular shape, which match both subchannel and sym-

bol divisions. Users' needs therefore have to be ceiled, which produces wasted slots.

We design a new mapping algorithm, sqTM, based on the Square treemap visualization algorithm of [68], which aims at reducing the amount of wasted slots. Though sqTM is slightly heavier than the reference algorithms we implemented, it manages to seriously reduce wasted slots, between 60% and 75% compared to reference algorithms.

In a full scheduling process, the reduced wastes of sqTM allow scheduling algorithm to use more efficiently radio resources, leading to higher cell throughputs, and lower packet delays. Our computer simulations showed sqTM requires 10 to 30 less subchannels than reference burst mapping algorithms to provide the same throughtput and packet delay performances.

Part IV

Conclusions

Chapter 10

Concluding remarks

The objective of this thesis was to study scheduling in cellular networks, and to propose some multistream enhancements. Indeed, the traditional scheme of scheduling users one after another has been shown suboptimal in many cases. This is more efficient to let multiple users transmit simultaneously, assuming generated interferences remain weak.

Because scheduling decisions have to be taken at a TTI scale of a few milliseconds, optimal and exhaustive computations can not be performed. We turned ourselves to heuristics, trying to balance throughput performances, user fairness, respect of QoS constraints, while limiting resource wasting.

Our first research framework was the current CDMA-based third generation UMTS network, and its HSPA evolution. Following standards evolution, we then refocused our work on OFDMA-based Mobile WiMAX and LTE frameworks. For both CDMA and OFDMA, we began with SISO transmissions, and considered MIMO beamforming transmissions afterwards.

10.1 CDMA-based networks

Current 3G networks use CDMA as their multiple access technique. Different simultaneous users' transmissions are therefore separated through orthogonal codes. If these transmissions are synchronized, which is straightforward in DL scenarios, there should be no interference. But actually, due to multipath effect, received transmissions are composed of multiple delayed echoes (taps) of the original transmissions, and are therefore impaired by interference.

The multistream scheduling issue is therefore the identification of simultaneously transmitting user groups with coupling due to these inter-

ferences. Indeed, each selected user interferes with all other transmitting users. We managed to model these interferences through an orthogonality factor β which depends, for each user, only on the power repartition of its received taps. Coupling is not an issue anymore, interference each user undergoes is defined by their own channel dispersion, and is independent of the selection of the other simultaneous users.

Based on this orthogonality factor, we introduced a multistream extension of the traditional PF algorithm, and compared it to a multistream RR and to several traditional single stream schedulers through computer simulations. Multistream throughput gain was especially obtained with low to moderate queue rates, when users can not benefit alone from the whole radio resources. Actually with high queue rates, multistream algorithms allocate all codes to the same user. Thanks to its PF criteria, our Multistream PF algorithm provides better fairness than the other algorithms, and since Multistream algorithms are able to schedule users more often, throughputs are smoother, less bursty, which can be an advantage for multimedia services such as audio and video streaming.

10.2 Beamforming in CDMA-based networks

We then considered MIMO techniques applied to CDMA-based networks. Among the different usages of multiple antennas, the one which seemed to us the most challenging from a scheduling point of view is beamforming. Beamforming consists of focusing transmissions into limited directions by the use of multiple phase-shifted antennas. The same codes can therefore be reused within each beam, provided beams are not overlapping. Interferences have therefore to be mitigated through a clever steering of the beams. We modeled these interferences through another orthogonality factor $G_{i\,k}$, based on beam radiation patterns.

But as far as scheduling is concerned, beams are not the same kind of resource as codes. Indeed, in MU-MISO with only one receive antenna per user, each user can not receive more than one beam, whereas it can benefit from all CDMA codes. Moreover, when it comes to steering another beam towards another user, the position of all simultaneous users determines the interference level. Beams can not be steered independently of one another. We therefore defined another heuristic algorithm suiting the beam allocation specificities.

Compared to a single full powered beam algorithm, our computer simulations showed some throughput gains, especially with large queue rates. Unlike SISO results, where large queue rates cause multistream

algorithms to allocate all codes to the same user, leading to no multistream gain, since with beamforming users can only have at most one beam, there is still multistream gain with large queue rates. Besides, fairness is also increased by the multistream beamforming algorithm.

10.3 GoB in CDMA-based networks

Since SISO computations, with multistream through orthogonal HSDPA codes, demonstrated multistream gain at low queue rates, while multistream through nearly orthogonal beams produced multistream gain at large queue rates, a natural question was about the combination of the two techniques.

For that purpose, we considered a GoB strategy. The idea of the GoB technique is to cover the cell sectors with fixed beams such as each user, depending on its location, is in the coverage area of one of the beams. As a result, within each beam, HSDPA codes can be allocated just as in SISO. Our simulations showed that the combination of the two multistream techniques is able to provide throughput gains with every queue rate. Gains on beamforming strategy were especially obtained at low queue rates, while larger queue rates allowed gains on SISO strategy. Fairness, however, is quite impaired by the grid design, which favors users closer to the center of their beam.

10.4 OFDMA-based networks

While 3G networks are based on CDMA, emerging 4G networks such as Mobile WiMAX and LTE consider OFDMA as their multiple access technique. Instead of orthogonal codes, simultaneous users are transmitting using different frequency bands. Thanks to longer symbols on narrower frequency bands, protected by cyclic prefix, there is ideally no ISI anymore. As a result, provided users transmit through disjoint frequency bands, there is no interference at all.

User selection is therefore simplified, but other challenges arise. Although already present in 3G, with different multimedia services, scheduling algorithms have to meet heterogeneous QoS requirements. Moreover, due to frequency selectivity, frequency bands are not interchangeable like CDMA codes. For each user, some frequency bands can be in good shape, while others are in deep fade. As a result, schedulers can be channel opportunistic, trying to benefit from frequency diversity, but they have to know channel quality users experience on tens of frequency bands, which

implies a tremendous feedback load, and then have to allocate users on their best bands.

To tackle this double issue of allocation complexity and feedback load burden, we considered averaged channel quality, as with the WiMAX PUSC and FUSC subcarrier gathering modes or with the LTE wideband CQI feedback mode. This costs the loss of frequency diversity gain, but it allowed us to design a simple 3-step hierarchical scheduler. It first orders sessions according to their QoS constraints, then assesses their resource needs according to their current average channel quality, and finally allocates them slots of resource thanks to a treemap visualization based algorithm. This is a quite simple algorithm, able to provide QoS guarantees, even within overloaded cells.

We compared it to two reference algorithms and a channel bound we set up, all benefitting from frequency diversity, in DL WiMAX, DL LTE and UL WiMAX frameworks, considering real traffic distribution data. In WiMAX, both DL and UL, our algorithm provides better throughput and packet delays when bandwidth is limited. But with larger bandwidth, frequency diversity of reference algorithms enables them to provide better throughput, however at the cost of a tremendous feedback load. In LTE, where the scheduling period is reduced from 5ms to 1ms, our algorithm provides poorer results than the reference algorithms, but with a seriously lower feedback load.

So finally, losing frequency selectivity can be more than compensated by an accurate allocation of radio resources when the scheduling period is long enough such in WiMAX scenarios. However with short scheduling period such as LTE TTI, resource wastes of opportunistic scheduler is no more an issue, and so our scheme is not the most efficient scheduler. Nevertheless, the reduced feedback load of a wideband CQI scheme such as ours is quite desirable, whatever the concerned standard is.

10.5 GoB in OFDMA-based networks

We then extended this OFDMA scheduler to MIMO through a GoB strategy. As in CDMA, GoB allows us to divide the coverage area into beams, within which we can share radio resources just as in SISO scenarios. Moreover, beamforming allows us to reuse frequency bands in non overlapping beams.

We therefore added a step gathering users into groups covered by the same beam before executing the same 3 steps of the SISO algorithm for each beam independently. Compared to a greedy scheduler steering beams towards effective users' position and benefitting from fre-

quency diversity, our algorithm is able to provide higher throughputs and lower delays in WiMAX. In LTE, due to shorter scheduling periods, performance of both algorithms are comparable. But still with both standards, considering averaged channel allows our algorithm to reduce tremendously the feedback load.

So such as in SISO, a wideband CQI scheme such as ours makes sense compared to opportunistic schedulers for long scheduling periods. Moreover, it seriously reduces feedback load.

10.6 WiMAX burst mapping

We finally dug more deeply into WiMAX resource allocation. There actually are some constraints we neglected earlier. Each allocation is specified by the DL- and UL-MAPs, and the more there are allocations, the larger those MAPs are. This control overhead can be minimized by gathering allocations using the same PHY-mode. Each allocation also must be of rectangular shape, matching both subchannel and symbol divisions, which imply ceiling wastes of resource. This burst mapping problem is actually a NP-HARD problem, to which only a few heuristics have been applied.

Based on the square treemap visualization algorithm, we designed sqTM, a new burst mapping algorithm, we compared to four reference algorithms. Our first comparisons, considering bursts of random size, showed an important reduction of slots wasted due to ceiling operations and MAP overhead, at the price of an increased but still acceptable computational time. We then evaluate our mapping gains on realistic scheduling computations. We considered our 3-step scheduler, and replace its mapping step successively by sqTM and our reference algorithms. The reduction of wasted slots of sqTM allows significative performance improvements, both on throughputs and packet delays. Using sqTM instead of the other algorithms allows larger throughputs and lower packet delays, from the same radio resources, or for the same throughputs and packet delays performance, it requires less resource than other burst mapping algorithms.

10.7 Perspectives

As far as future work is concerned, since CDMA is no more considered by emerging 4G networks, further investigations should obviously focus on OFDMA environments, especially considering MIMO techniques, for

their benefits on throughput, transmission reliability and resource reuse factor.

Going to wider and wider frequency bands, the issue of feedback load is ever increasing. A lot of existing scheduling algorithms now promise high spectral efficiency and throughput performances but neglect the impact of their feedback requirements. In the case of DL transmissions, transmitting feedback to base stations is an important burden for user equipments, which can drain their batteries. As we have shown, the feedback cost of frequency diversity gain finally seems pretty high for the benefits it allows, and future work should focus on reducing feedback requirements.

With the widespreading of smartphones and the ever increasing demand for diverse multimedia services, the importance of guaranteeing heterogeneous QoS requirements is getting more and more important. Though QoS guarantees are already well studied, setting up QoS requirements that accurately match quality of experience users are willing to have, for new and future multimedia services, seems to be quite important too when evaluating scheduling algorithms.

Our work also pointed out the importance of how resources are allocated, and the resulting waste of resource. The same scheduling algorithm, depending on the burst mapping algorithm it relies on, can provide very different throughputs and packet delays results. While scheduling and user selection is a well investigated topic, more efficiency gains can probably now be obtained from cleverer mapping techniques, as we have demonstrated in DL WiMAX scenarios. In the same way, resource allocation in LTE and LTE-Advanced can likely be optimized and should be further investigated.

Part V

Appendices

Appendix A

Peer-reviewed publications

A.1 Multistream Proportional Fair Packet Scheduling Optimization in HSDPA

Presented at the 15th Annual Symposium of the IEEE/CVT Benelux Chapter (SCVT). Antwerpen (Belgium), november 2008.

A.2 Treemap-based Burst Mapping Algorithm for Downlink Mobile WiMAX Systems

Presented at the IEEE 74th Vehicular Technology Conference (VTC-Fall). San Francisco (United States), september 2011.

Multistream Proportional Fair Packet Scheduling Optimization in HSDPA

J. Vanderpypen and L. Schumacher, FUNDP – The University of Namur, {jva,lsc}@info.fundp.ac.be

***Abstract*—To improve wireless channel capacity, one can reduce radio resource waste through the scheduling of multiple users within a TTI. In interference-limited systems, user selection should be performed carefully, so as to limit interference. In this paper, we present a multistream scheduler, based on a Proportional Fair strategy, for its trade-off between fairness among users and cell throughput. This scheduler is a refinement of an earlier proposal, where the queue length of each user is taken into account, to improve the overall resource efficiency. To avoid significant computations, a heuristic has been developed. The scheduler performance is compared against a multistream Round Robin scheduler and four traditional single user schedulers via computer simulations, in the peculiar case of HSDPA transmissions. Both throughput and fairness gain are highlighted, especially with low users queue rates. A 8% gain in total cell throughout is achieved, while preserving fairness.**

***Index Terms*—Multistream scheduling, Proportional Fair, HSDPA.**

I. Introduction

In a cellular network, all users of a given cell are connected to the same base station, and the transmissions of all users are controlled by a scheduling algorithm. For each Time Transmit Interval (TTI), the scheduler shares the transmission resources, which are transmit power, spreading codes, frequency bands, LTE's Physical Resource Blocks (PRB) and so on. In interference-limited high speed cellular networks like HSDPA (High Speed Downlink Packet Access), the standard prescribes that a single user is scheduled in each cell for a TTI. Actually, it has been shown in [1] that this approach is suboptimal because when channel quality is poor, the scheduled user can not fully benefit from the allocated resource. So it would be more efficient to share the available resources among several users. As [2] mentions, the selection of these simultaneous users should be done carefully, to limit interference between each other.

There are many scheduling algorithms. The simplest one is the Round Robin (RR) scheduler, where each user is scheduled at a periodic time slot. This algorithm is really fair between users, but its throughput is not optimal. Actually, it is more efficient to schedule users when their variable channel is in good condition, i.e. when users would benefit the most from the resources they have been allocated. Another scheduling strategy, called Max Throughput, is therefore to schedule the user who has the best channel. The trouble with this technique is fairness among users: a distant user will never get scheduled as long as there are users with better channel conditions. There are also some algorithms taking QoS into account, as the Max Weight scheduler for example. It can offer some guarantees on delay and delay jitter thanks to its criteria, which is the potential throughput weighted by the queue length. To achieve a good trade-off between fairness among users and cell throughput, we chose the Proportional Fair algorithm, where the criterium to maximize is also the potential throughput of the user, but normalized by its mean throughput so far.

This paper will detail the behaviour of HSDPA-enabled cellular networks. In HSDPA, there are 15 physical channels, determined by orthogonal channelisation codes, that are shared between active users. Selecting exhaustively the best group of simultaneous users, and the optimal code repartition implies heavy computations every TTI. Let us point out that in HSDPA, a TTI only lasts 2 ms [3]. This is the reason why we looked for a heuristic.

The algorithm we propose is based on a Proportional Fair criterium, where we originally introduce some QoS parameter. The algorithm is split into two phases: a first phase consists of identifying the best users to schedule, and the second one allocates them the radio resources.

The rest of the paper is organized as follows. Section II presents the system and the channel model. Our multistream scheduling algorithm is detailed in Section III, and its performance are evaluated within Section IV. Finally, Section V draws some conclusions about this work.

II. System Model

Since our scheduling criteria is based on Proportional Fair, it requires knowledge on potential throughput and mean throughput. It is quite easy for the base station to log the throughputs of its users and evaluate their mean throughput. But as far as the potential throughput is concerned, it needs to be estimated, depending on resource allocation. So our first step is to get a rough estimate of the achievable bit rate of each user in the next TTI. This rate is upper bounded by the Shannon capacity [4]:

$$C_k = BW \cdot \log_2\left(1 + SNIR_k\right) \tag{1}$$

where the channel capacity of the user $k = 1, 2, \ldots, N_{Queued}$ depends on the bandwidth *BW* and the Signal to Noise plus Interference Ratio *SNIR* of its channel. These channel capacities are then truncated with respect to the queue length Q_k.

Hence, the potentially achievable throughput B_k is given by:

$$B_k = min\left(C_k, Q_k / TTI\right) \tag{2}$$

*J. Vanderpypen acknowledges the funding support of the "Fonds à la formation pour la Recheche dans l'Industrie et dans l'Agriculture" (F.R.I.A.).

This truncation is an original amendment to the algorithm proposed in [5]. The bound Q_k / TTI actually represents the bit rate which would empty the queue Q_k within the duration of one TTI. Indeed, even with a large channel capacity, if a user only has a short queue, it is worthless to over-estimate his/her throughput.

The received signal power S_k of a user k depends on the channel state $|H_k|$, his/her transmit power P_k, and his/her path loss d_k^γ, which represents the signal attenuation due to the distance d_k between the base station and the user k. The received signal power is obtained from

$$S_k = |H_k|^2 \frac{P_k}{d_k^\gamma} \quad (3)$$

The thermal noise is a constant. As far as interference are considered, because HSDPA is based on orthogonal codes, there is no interference between transmissions of a cell in the downlink as long as these transmissions are synchronized. However, because of multipath fading, several echoes of each transmitted signal will be received, and will bring interference into the system. The proportion of the transmitted signal which will be perceived as interference can be estimated by the orthogonality factor β [6]:

$$\beta = 1 - \frac{\sum_{i=1}^{L} (\alpha_i)^4}{\left(\sum_{i=1}^{L} (\alpha_i)^2\right)^2} \; ; \quad (4)$$

where L is the number of taps modelling the channel dispersion, and (α_i) is the power of the i^{th} one. One can see that if there is only one significant echo, we have $\beta \approx 0$, so there is no interference.

As a result, the SNIR of a given user k can be estimated by

$$SNIR_k = \frac{|H_k|^2 \frac{P_k}{d_k^\gamma}}{\sigma_k + |H_k|^2 \frac{\beta_k P_{max}}{d_k^\gamma}} \quad (5)$$

where σ_k represents the noise power, and P_{max} the total emitting power of the cell. Once the SNIR is known for each user, all PF criteria can be obtained from (2), and scheduling choices can be done. These choices will be detailed within the next section.

III. Scheduling Algorithm

The criteria we will use to compare the users is based on Proportional Fair, to achieve a good trade-off between fairness among users and cell throughput. It consists of maximizing B_k, the potential throughput of the user k, weighted by $\overline{B_k}$, its mean throughput:

$$\frac{B_k}{\overline{B_k}} \quad (6)$$

So the scheduled user will have a high throughput thanks to B_k, and the fairness is achieved by the mean throughput weighting. For example, if a user has constantly a good channel and backlogged data to transmit, s/he will initially be frequently scheduled. But after some time, his/her mean throughput will increase, and the ratio in (6) will get lower than the one of another user with poorer channel quality which still has not transmitted yet.

To find the optimal resource allocation, we should evaluate this Proportional Fair criteria for all possible groups of scheduled users, for all possible code repartition between them. This 2-dimensional exhaustive search, in user and code spaces, can not be performed within a TTI. Only for the user selection, with 15 HSDPA codes, the total number of possible user groups is

$$\sum_{k=1}^{15} C_{N_{queued}}^k . \quad (7)$$

And then for each group, we should test all code combinations. Hence, our interest for a heuristic.

In [5], it is suggested to first select users without taking into account the interference, so this selection can be performed linearly. When the scheduled users are known, the next step is to share the resource among them.

So our first step is to evaluate the potential throughput each user would obtain from one radio resource quantum, which can be a frequency band, a PRB, or a set of orthogonal codes, with the corresponding fraction of the transmit power. In our case of HSDPA transmissions, where there are maximum 15 codes to share and no power control, a quantum will be a single code, powered with 1/15 of the transmit power devoted to HSDPA.

Actually, with CDMA-based transmissions, like HSDPA, multi-user interference only depends on the transmit powers involved. Indeed, since all users use the same spreading factor, the interfering effect of a given code does not depend on the peculiar user whom the code belongs; multiple access interference (MAI) and intersymbol interference (ISI) are mistaken within each other

So if we consider that all 15 codes are used, we can estimate the throughput each user can get from one code, with 1/15 of the power P_{HSDPA} allocated to HSDPA transmissions. And this estimation can be done without any knowledge on the set of the other simultaneous users. Therefore, our second step is a loop on the resource: while there are still available codes, find the user which would maximize his/her PF criteria and allocate him/her the number of codes s/he needs. Obviously, we can not allocate a fraction of code. The number of codes are therefore rounded to the next integer towards zero, unless rounding leads to zero. A user should have at least one code to transmit, but we want to avoid wasting resource.

The multistream Proportional Fair scheduler can be synthesized in the following algorithm (I). To evaluate the multistream gain, we compared the new scheduler with a multistream Round Robin and four traditional mono-user schedulers: Round Robin, Max Throughput, Proportional Fair, and Max Weight.

1. Evaluate the throughput each user would benefit from a single code with 1/15 of the HSDPA transmit power:

$$\hat{B}_k = min\left\{BW \cdot \log_2\left(1+SNIR_k\right),\ Q_k/TTI\right\},\quad \forall k=1,\ldots,N_{Queued} \tag{8}$$

where

$$SNIR_k = \frac{\left|\overline{H_k}\right|^2 \frac{P_{HSDPA}/15}{d_k^{\gamma}}}{\sigma_k + \left|\overline{H_k}\right|^2 \frac{\beta_k P_{max}}{d_k^{\gamma}}} \tag{9}$$

2. While there are still available codes, find the user which would benefit the most from them:
   1. For all user $k=1,\ldots,N_{Queued}$:
      - Evaluate the number of codes they need to maximize their throughput (i.e. empty their queue):

$$nc_k = \left\lceil \frac{Q_k}{\hat{B}_k \cdot TTI} \right\rceil - 1; \tag{10}$$

      where $\lceil x \rceil$ is the smallest integer bigger than *x*.
      - Truncate this number of codes by the number of codes still available;
      - Evaluate the PF criteria of each user:

$$PF_k = \frac{min\left\{nc_k \cdot \hat{B}_k,\ Q_k/TTI\right\}}{\overline{B_k}}; \tag{11}$$

   2. Attribute $nc_{\tilde{k}}$ codes to the user $\tilde{k} = \underset{k=1,\ldots,N_{Queued}}{arg\,max}\ PF_k$.

Algorithm (I) : Multistream Proportional Fair scheduling algorithm.

The Round Robin scheduler selects each user at a periodic time slot, and the multistream Round Robin also allocates resources in a carrousel kind of way, but if the selected user can not benefit from all codes, the unused ones can be allocated to the following users.

The Max Throughput chooses the user i_{MT} which has the maximum throughput:

$$i_{MT} = \underset{k=1,\ldots,K}{arg\,max}\ B_k \tag{12}$$

The traditional Proportional Fair selects the user i_{PF} that maximizes

$$i_{PF} = \underset{k=1,\ldots,K}{arg\,max}\ \frac{B_k}{\overline{B_k}} \tag{13}$$

And finally, the Max Weight scheduler select the user i_{MW} that maximizes

$$i_{MW} = \underset{k=1,\ldots,K}{arg\,max}\ B_k \cdot Q_k \tag{14}$$

As a matter of fair comparison, all these schedulers have relied on the bounded throughout (2).The next section will now evaluate the performance of the multistream scheduler.

IV. MATLAB SIMULATIONS

To evaluate the performance of the new scheduler, we did some computer simulations in MATLAB. We used SCME, a channel model implemented by the IST-WINNER project [7] according to the 3GPP standardization group specifications.

We considered 8 users uniformly distributed in a ring, between 35 and 500m away from the base station, moving with a velocity of 10 m/s. The bandwidth of the system is 5 MHz, and the carrier frequency is 2 GHz. As [8, Table 12.7] mentions, the maximum transmit power dedicated to HSDPA is 45 dBm, and the thermal noise power is -101.2 dBm.

To model the queue rates, we considered a fixed arrival packet rate of 1 packet per TTI, and a uniformly distributed packet size. With a packet size uniformly distributed in [1.5 kb, 4.5 kb], we get an average queue rate of 1,500 kbps per user, which means that the cell has to transmit for the 8 users in average 2,400 kb during the 100 TTI's of our simulations. We also considered twice and half of this queue rate.

Following figures compare the cumulative probability density function of the cell throughput for the different schedulers, with different mean queue rates. Results are averaged over 200 runs of 100 TTIs each. Fig. 1 considers a low mean queue rate of 750 kbps for each user, Fig. 2 1,500 kbps, and Fig. 3 3,000 kbps.

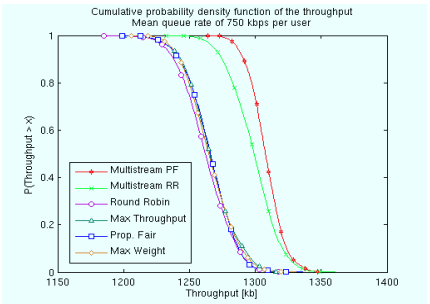


Fig. 1 : Cumulative probability density function of the cell throughput, for 8 users with a mean queue rate of 750 kbps each.

As one can see on Fig. 1, with such low queue rates, all traditional schedulers lead to the same throughput, whereas the multistream schedulers achieve a gain around 100 kb during the 100 TTI's of the run, which means 8% of gain. As one can see, because the multistream PF takes the channel conditions into account to select users, it has slightly better performance.

Fig. 2 shows that with doubled queue rates, throughput disparities appear between single user schedulers. Round Robin lead to the lowest throughput, whereas Max Weight and Max Throughput have the highest throughput among the traditional schedulers. As far as multistream scheduling is concerned, one can see that it leads to the best throughputs, even better than the Max Throughput, but no difference really appears between Proportional Fair and

Round Robin strategies. The gain on the traditional Proportional Fair scheduler is 8%, and there is still around 6% gain on Max Throughput. And since our multistream scheduler is based on the Proportional Fair strategy, we can state that it is also fairer than Max Throughput.

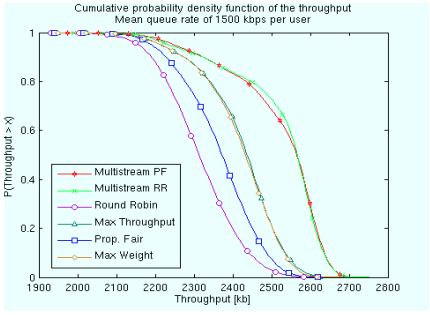


Fig. 2 : Cumulative probability density function of the cell throughput, for 8 users with a mean queue rate of 1500 kbps each.

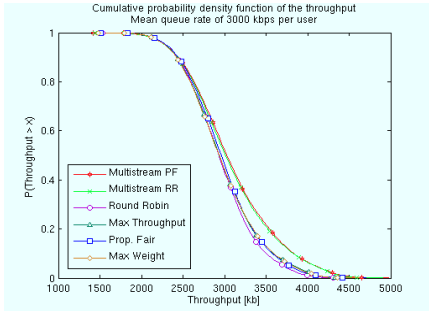


Fig. 3 : Cumulative probability density function of the cell throughput, for 8 users with a mean queue rate of 3000 kbps each.

Finally, as Fig. 3 shows, when queue rates are really large, every user has sufficient data in his/her queue to plainly benefit from all codes. Therefore all schedulers lead to the same throughput, limited by the cell capacity. Indeed, as Figs. 4 & 5 show, with the highest queue rate, most of the time only one user is scheduled within a TTI, such that multistream scheduling is pointless. On the other hand, lower queue rates lead to multiple simultaneous users, up to 8 90% of the time for the smallest queue rate. Indeed, multiple user scheduling is only preferred when a single user can not benefit alone from all radio resources.

To compare the distribution of the number of simultaneous users of the two multistream schedulers, one can say from Figs. 4 & 5 that they behave the same at low and high queue rates. A difference only appears at moderate queue rates, where the multistream Proportional Fair schedules a single user 50% of the time. The multistream Round Robin selects more often multiple users, as it is shown on Fig. 5.

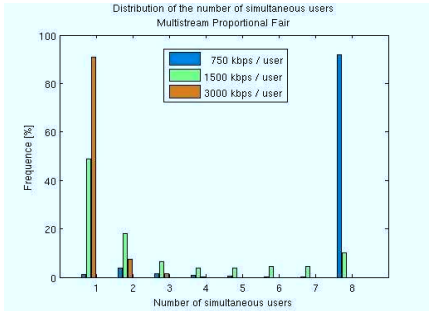


Fig. 4 : Distribution of the number of simultaneous users, for the different queue rates with the multistream PF scheduler

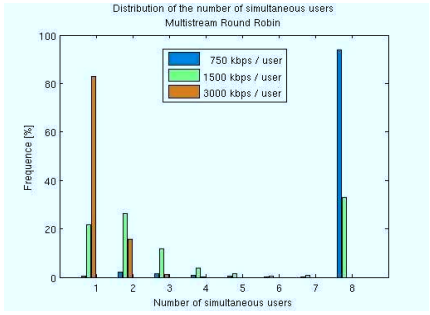


Fig. 5 : Distribution of the number of simultaneous users, for the different queue rates with the multistream RR scheduler

We also tried to double the number of users, each with half of the studied queue rates. The behaviour and the gains are similar: high queue rates lead to no multistream gain whereas low queue rates lead to 6-8% of throughput gain, because low queue rates lead to multiple simultaneous users (15 users 90% of the time with in average 375 kbps for each user queue rate).

As far as fairness is concerned, we evaluated the Jain Index [9]. The advantage of the Jain index is that it is bounded in between $1/N_{Queued}$ and 1, dimensionless and independent of the number of users. If we write $\tilde{B}_k$ the actual transferred throughput of the user k at the end of the simulation, the Jain Index is defined by

$$Jain\ Index \equiv \frac{\left(\sum_{k=1}^{N_{Queued}} \tilde{B}_k\right)^2}{N_{Queued} \cdot \sum_{k=1}^{N_{Queued}} \left(\tilde{B}_k\right)^2}. \qquad (15)$$

We evaluated the Jain Index of the different schedulers and averaged the results over 2,000 runs of 25 TTIs each. Only 25 TTIs have been considered to avoid long term averaging, where fairness is actually determined by the same queue rate of each user.

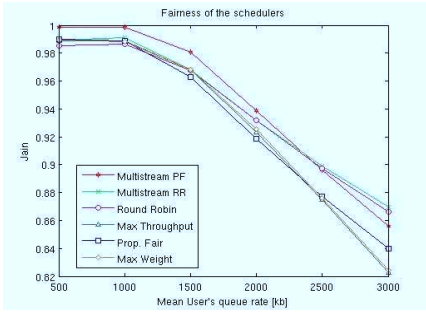


Fig. 6 : Fairness of the different schedulers, depending on the user's queue rate

As one can see on Fig. 6, at low queue rates, all schedulers are as fair as each other, except the multistream Proportional Fair, which is slightly fairer. At moderate queue rates, multistream Proportional Fair becomes as fair as single user Round Robin, but with better throughput. Finally, at high queue rates, multistream Proportional Fair becomes a little less fair than single user Round Robin, but still fairer than single stream Proportional Fair. Max Throughput and Max Weight, which produce the same throughput, lead to equal fairness. As far as Round Robin is concerned, multistream does not seems to really improve fairness, even if the throughout gain is obvious. A strange behavior is that multistream Proportional Fair has better fairness performance than multistream Round Robin for low to moderate queue rates, while they have quite the same cell throughout. This probably comes from the Proportional Fair criteria, which takes into account the queue length and the channel capacity of each user. As a result, users are selected in order to have similar throughput with the multistream Proportional Fair, whereas multistream Round Robin is only fair on the number of TTIs each user has been scheduled.

We also evaluated the fairness after only 10 TTIs, and schedulers behave similarly.

Finally, another advantage of multistream scheduling is that all users can transmit more often than with single stream schedulers, so their throughput is smoother, less bursty. It effect can be a real advantage for 3G multimedia services, like video streaming for example.

V. Conclusions

Since scheduling a single user within a TTI can lead to radio resource waste, we introduced a multistream scheduler. For a good trade-off between fairness among users and throughput, we based our work on a Proportional Fair-like criteria, but actually the criteria can be changed for any other one. The scheduling algorithm has been set up to minimize interference between simultaneous users. We focused on HSDPA transmissions, and evaluate both throughput and fairness performance of the multistream scheduler via computer simulations. Those performance have been compared with four traditional single user schedulers, namely Round Robin, Max Throughput, Proportional Fair and Max Weight, and both throughput and fairness gain has been observed, particularly for low to moderate queue rates. Indeed, when queue rates are too high, a single user can benefit from all radio resource alone, and therefore most of the time the multistream scheduler only select one user per TTI.

For future work, we should also consider Multiple-Input Multiple-Output (MIMO) transmissions, and deal with antenna sharing. MIMO techniques are really promising, and would be able to make transmissions more reliable [10], to bring better throughput [11], and to restrict the antenna emissions to limited directions [12]. Because our throughputs are only estimated by the Shannon bound, it would also be interesting to consider the effects of a SNIR depending coding scheme.

References

[1] R. Kwan, P. Chong, E. Poutiainen et M. Rinne, The Effect of Code-Multiplexing on the HSDPA in a WBCDMA Network, in Proceedings of IEEE WCNC'03 conference, 2003
[2] L. T. Berger, Performance of Multi-Antenna Enhanced HSDPA, PhD Thesis, Aalborg, 2005
[3] 3GPP TR 25.814 v7.1.0, Physical layer aspects for evolved Universal Terrestrial Radio Access, September 2006
[4] C. E. Shannon, A Mathematical Theory of Comunication, in Bell Sys. Tech. Journal, 27, 1948
[5] S. Vangipuram et S. Bhashyam, Multiuser Scheduling and Power Sharing for CDMA Packet Data Systems, in Proceedings of IEEE NCC Conference, 2007
[6] N. Mehta, A. Molisch et L. Greenstein, Orthogonality Factor in WCDMA Downlinks in Urban Macrocellular Environments, in Proceedings of IEEE Globecom 2005 Conference, 2005
[7] 6th Framework Programme, Information Society Technologies, Wireless World Initiative New Radio (WINNER), IST-2003-507591, https://www.ist-winner.org
[8] Holma and A. Toskala, WCDMA For UMTS - HSDPA Evolutions and LTE, 2007
[9] R. Jain, D.M. Chiu, and W. Hawe, A Quantitative Measure of Fairness and Discrimination for Resource Allocation in Shared Systems. DEC Research Report TR-301. 1984
[10] H. Schulze et C. Lüders, Theory and application of OFDM and CDMA, in John Wiley & Sons, 2005
[11] I. E. Telatar, Capacity of multi-antenna gaussian channels, in European Transactions on Telecommunications, vol. 10, pp. 585–595, Nov./Dec. 1999
[12] P. Stoica et R. Moses, Spectral Analysis of Signal, in Pearson Prentice Hall, 2005

Treemap-based Burst Mapping Algorithm for Downlink Mobile WiMAX Systems

Joël Vanderpypen, Laurent Schumacher
FUNDP - The University of Namur, Belgium
Computer Science Faculty
{jva, lsc} @ info.fundp.ac.be

***Abstract*—This paper presents our sqTM burst mapping algorithm for downlink Mobile WiMAX systems. Based on a treemap visualization algorithm, we introduce a new burst mapping scheme we called sqTM, greatly reducing the amount of wasted slots. We obtained between 60% and 75% reduction compared to some reference algorithms we implemented. These limited wastes are able to provide better cell throughput and larger cell capacities. Unfortunately, sqTM is significantly slower than reference algorithms, but still easily coping with 5 ms frames.**

I. Introduction

Mobile WiMAX [1] has been pushed forwards by the IEEE as the standard for Broadband Wireless Access. It is based on Orthogonal Frequency Division Multiple Access (OFDMA), so the frequency band is divided into many subcarriers. Considering Partial Usage of the SubChannels (PUSC), which is the most common mode [2], the subcarriers are logically permuted before being gathered into subchannels. As a result, intercell interferences and the effect of fast fading are reduced.

The radio resource is the frame, which represent the usage of several subchannels for a 5 ms duration. A frame is divided into two parts: the downlink (DL) and the uplink (UL) subframes. In Time Division Duplexing (TDD), DL and UL subframes are time alternated, while in Frequency Division Duplexing (FDD), they use different frequency bands.

Resource allocation is performed by slots, which represent the usage of one subchannel for one symbol duration. Depending of the channel quality, a given user will use a specific PHY-profile, which determines the throughput he/she benefits from a slot. The allocation is done per burst, where a burst contains the data of one or multiple users, as long as a single PHY-mode is used per burst.

The burst mapping problem is quite different in DL and UL. After a one symbol duration preamble, the DL subframe is composed of a Frame Control Header (FCH) and the DL- and UL-MAPs, which are maps specifying the burst profile of each allocation. Then each burst receives a rectangle allocation, whose shape and position are specified through the Information Elements (IEs) of the DL-MAP. The more there are bursts, and the bigger the DL-MAP is. So it is preferable to gather users into a minimal number of bursts, the unique connection identifier (CID) separating them. In UL, the bursts are simply mapped as contiguous slots in a row wise order.

J. Vanderpypen acknowledges the funding support of the Fonds à la formation pour la Recherche dans l'Industrie et dans l'Agriculture (F.R.I.A.)

Since the UL mapping problem is straightforward, we focus here on DL. This tiling problem is actually a bi-dimensional bin packing problem, a variation of the knapsack problem, which has been proven NP-Hard [3]. The authors of [4] presented a heuristic where bursts size are rounded to a multiple of the height of the frame, leading to a consequent waste of slots. There are other strategies like SDRA [5] or the Raster-based algorithm [6], with virtually no wasted slots, but they multiply the number of bursts, and therefore increase significantly the DL-MAP signaling overhead. The authors of [7], [8] came up with some variations of what they called the Sequential Rectangle Placement (SRP). They define three sizes of bursts, and gather them into vertical stripes of growing width, according to their size category. The OCSA algorithm [9], [10] introduces an other stripe gathering strategy, more complex, but leading to fewer wasted slots.

In this paper, we consider a treemap visualization algorithm [11], and tune it to meet the WiMAX requirements to propose a more efficient heuristic named *sqTM*. Like OCSA, we gather bursts into vertical stripes, but of growing width, trying to limit the amount of wasted slots. As a result, our algorithm is able to seriously reduce the waste of slots while not increasing the DL-MAP. The rest of this paper is organized as follows. Treemap visualization is first introduced in Section II, then our sqTM algorithm is explained in Section III. Section IV details the reference algorithms we implemented into MATLAB scripts. Afterwards some computer simulation results are presented in Section V, first to compare algorithms on a single snapshot allocation, then to present averaged results. Finally some conclusions are drawn in Section VI.

II. Treemap visualization

Treemap visualization has been initially designed to visualize hard disk usage, in order to ease the identification of large files. The tree structure of a file system is represented by rectangles fitted into each other, whose area is proportional to the disk usage of the directory they represent.

A simple algorithm presented in [12] lists the size of all directories at the root level and divides the original rectangle representing the whole disk into different slices, whose width is proportional to the size of all first level directories. A special slice to represent free space has to be added. And then, recursively, for each directory, its area is cut into smaller slices representing all its sub-directories. The cuts should be done

alternatively horizontally and vertically. One drawback of this visualization algorithm is that it can lead to thin rectangles, whereas one would prefer rectangles with aspect ratio close to one (nearly square rectangles), for a better visualization.

The authors of [11] present a Squared Treemap algorithm, dividing a rectangle into smaller nearly squared rectangles, by cutting both horizontally and vertically. The idea is to cut a first vertical stripe into the original rectangle, whose area represents the first directory. Then an other directory is added to that stripe, which therefore becomes wider. The different directories of a stripe are divided by horizontal cuts. And so on, other directories are added to that stripe, until their aspect ratio can not be improved anymore. Then other vertical stripes are cut and filled with the remaining directories. This algorithm is a heuristic of linear time complexity.

III. Our sqTM algorithm

Even if we are not interested by the aspect ratio of the bursts, the procedure of the Squared Treemap algorithm seemed us pretty useful for the burst mapping problem. However, its application is not straightforward. Indeed, different bursts can not share the same subchannel during the same symbol time. So ceiling operations have to be considered, both at subcarrier scale on frequency dimension and at symbol scale on time dimension. These ceiling operations lead to wasted slots, whose quantity has to be minimized. Our allocation criteria is therefore not the aspect ratio, but the number of wasted slots.

A. The allocation procedure

Bursts are treated sequentially, and do not need to be sorted out. The allocation is done through vertical stripes of variable width. Each stripe is first composed of a single burst, with full height and rounded width. The number of wasted slots is evaluated. Then we sequentially try to add each of the unallocated bursts. If adding the burst reduces the number of wasted slots, it is added to the stripe. When adding any of the remaining bursts can not reduce the number of wasted slots, we start a new stripe, and try each unallocated burst.

The first stripe is built to include the FCH and the DL-MAP.

B. Ceiling operation

This section details how the resources are allocated to a stripe, and shared to its different bursts. Let us consider a stripe R composed of bursts b_i with $i = 1, ..., n_B$, each burst requiring r_i slots. The height of the frame is denoted by H.

Firstly, an accurate allocation is computed, with no ceiling, as Fig. 1(a) shows. The temporary width of the stripe $\tilde{w}_R$ is obtained as follows:

$$\tilde{w}_R = \frac{\sum_{b_i \in R} r_i}{H} \tag{1}$$

and the temporary height $\tilde{h}_i$ of each burst is given by

$$\tilde{h}_i = \frac{r_i}{\tilde{w}_R} \quad \forall i = 1, \cdots, n_B. \tag{2}$$

Secondly, the height allocated to each burst has to be rounded to h_i, to match subchannel scale, paying attention that the sum of all rounded heights must be equal to the height of the frame. Defining the operator $\lceil \cdot \rfloor$ such as it rounds to the closest integer, we have:

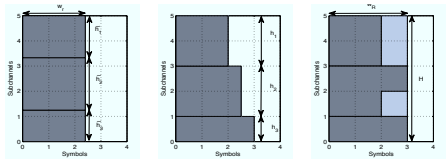
$$h_i = \begin{cases} \left\lceil \tilde{h}_i \right\rfloor & \forall i = 1, ..., n_B - 1; \\ H - \sum_{j=1}^{n_B - 1} h_j & i = n_B. \end{cases} \tag{3}$$

As shown on Fig. 1(b), the width of each burst has to be adapted consequently to $\tilde{\tilde{w}}_i$:

$$\tilde{\tilde{w}}_i = \frac{r_i}{h_i} \quad \forall i = 1, \cdots, n_B. \tag{4}$$

And thirdly, the width of the stripe w_R, which is also the final width w_i of each burst, can be fixed to the closest larger integer:

$$w_R = \max_{i=1,\cdots,n_B} \left\lceil \tilde{\tilde{w}}_i \right\rceil. \tag{5}$$



(a) Accurate allocation (b) Rounded heights (c) Ceilled allocation

Fig. 1. Ceiling operation, with bursts of size 4, 5 and 3 slots. After ceiling, 3 slots are wasted.

The number t_{wasted} of slots wasted in the stripe allocation of Fig. 1(c) can be estimated by

$$t_{wasted} = (H \cdot w_R) - \sum_{i=1}^{n_B} r_i. \tag{6}$$

To limit computations, we can perform a single test before the full ceiling operation, which is only fruitful if adding the new burst reduces the number of wasted slots. After computing (1), even if rounded heights shown on Fig. 1(b) would not lead to any waste, we loose at least $H \cdot (\lceil \tilde{w}_R \rceil - \tilde{w}_R)$ slots. This is a minimum bound on the amount of wasted slots for the stripe. If this bound is greater than the amount of wasted slots of a previous stripe composition, other ceiling computations are useless, we know the current stripe composition does not produce better results than the previous one.

C. Algorithm

We therefore came up with the Algorithm 1. While there are still unallocated bursts, it makes a new stripe, sequentially trying to add all the unallocated bursts. If the stripe is empty, the burst is directly added to the stripe (lines 4 to 6). If not, the ceiling of the previous section has to be done. The test of line 9 prevents the algorithm to perform some computations (lines 10 to 12, plus the evaluation of line 13) when we know they are pointless. Finally, after all computations, if wasted slots are reduced by adding burst b_i (line13), it is definitely added to the stripe, and other bursts are then tried to be added.

Algorithm 1 sqTM

while some unallocated bursts remain **do**
  Start a new stripe R;
  for each unallocated burst b_i **do**
    if stripe R is empty **then**
      Add b_i into R;
      $t_{wasted} \leftarrow (H - r_i \bmod H)$
    else {R is non empty}
      $\tilde{w}_R \leftarrow \sum_{b_j \in R \cup \{b_i\}} r_j / H$;
      if $(\lceil \tilde{w}_R \rceil - \tilde{w}_R) \cdot H < t_{wasted}$ **then**
        $h_j \leftarrow \lceil r_j / \tilde{w}_R \rfloor \quad \forall b_j \in R$;
        $h_i \leftarrow H - \sum_{b_j \in R} h_j$;
        $w_R \leftarrow \max_{b_j \in R \cup \{b_i\}} \lceil r_j / w_R \rceil$;
        if $(H \cdot w_R) - \sum_{b_j \in R \cup \{b_i\}} r_j \leq t_{wasted}$ **then**
          Add b_i into R;
          $t_{wasted} \leftarrow (H \cdot w_R) - \sum_{b_j \in R \cup \{b_i\}} r_j$;
        end if
      end if
    end if
  end for
end while

D. Complexity

The algorithm is composed of a main while loop, limited by the number n of bursts. Then for each stripe, all unallocated bursts are tested, which number is also bounded by n. As a result, the complexity of our sqTM algorithm is $\mathcal{O}(n^2)$.

IV. Some reference algorithms

To evaluate the performance of sqTM, we have implemented four other burst mapping schemes, namely the bucket-based algorithm [4], SDRA [5], SRP [7] and OCSA [9].

A. Bucket-based

The authors of [4] proposed a simple algorithm, where burst sizes are ceiled to a multiple of H, the number of subchannels, and are handled sequentially. Bursts receive as many one-symbol width columns as they require. This algorithm is of complexity $\mathcal{O}(n)$. In average, it wastes $\frac{1+H}{2}$ slots per burst since the last column of each burst can contain 1 to H slots.

B. SDRA

The SDRA algorithm of [5] has the same column filling strategy as the bucket-based algorithm, but here when the allocation of one burst is done, it continues filling the same one-symbol width column with the allocation of the next burst. As a result, no slots are wasted by any ceiling operation, and this algorithm is also of complexity $\mathcal{O}(n)$.

However, to meet the rectangle shape allocation, each burst actually receive multiple rectangles. Depending on its size, each burst can receive up to 3 rectangles: one to complete the previous column, one with several full columns, and one for the end of the allocation on a new column. Because the size of the DL-MAP depends on the number of rectangle allocations, even if this algorithm does not waste any slot by ceiling operations, many slots are wasted on DL-MAP signaling.

C. SRP

The authors of [7] modeled the resource allocation problem as a Sequential Rectangle Placement. They defined three classes of bursts, with W standing for the number of symbols of the downlink scheduling period:

- the small bursts, with $r_i \leq 2\sqrt{H}$. These ones can only have one-symbol width allocation.
- the medium bursts, with $2\sqrt{H} < r_i \leq \frac{W}{2}\sqrt{H}$, receiving maximum $\frac{W}{2}$-symbol width allocation.
- the large bursts, with $r_i > \frac{W}{2}\sqrt{H}$, with maximum $2W$-symbol width allocation.

They also defined the concept of job sets, which are column shaped and contain only bursts of the same kind. The maximum width of a job set R is the maximum width of its bursts, and is denoted by $\text{MAX}(R)$. The width of a job set R is denoted $\text{WIDTH}(R)$ and is evaluated by:

$$\text{WIDTH}(R) = \min \left\{ \left\lceil \frac{\sum_{b_j \in R} r_j}{H - |R|} \right\rceil \ , \ \text{MAX}(R) \right\} \quad (7)$$

They sequentially handle each burst, trying to add it within the current job set of its size. If there remains not enough room, they close the job set, and start a new one. So SRP complexity is $\mathcal{O}(n)$, even if it requires more computations than the two previous algorithms.

D. OCSA

The authors of [9], [10] presented an algorithm where bursts have to be sorted in a descending order (largest area first). The largest unallocated burst is placed first, in a vertical stripe as narrow as possible. Then on the top of that burst, the algorithm tries to place the maximum amount of bursts fitting in the stripe without modifying its width. When no more burst can be added on the top of it, an other stripe is started with the largest unallocated burst remaining, and so on.

Since there can be n stripes, and for each one the n bursts can be tested, the total complexity of OCSA is $\mathcal{O}(n^2)$.

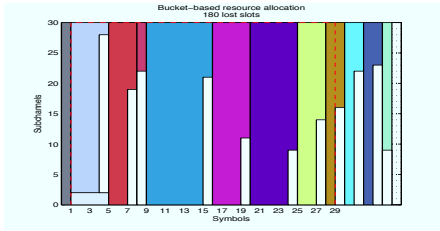
V. Performance evaluation

We implemented both our sqTM algorithm and the four reference ones into MATLAB scripts. We will detail first the parameters of these simulations, then we will present some allocation results. Afterwards, averaged results on the amount of wasted slots and computational time will be presented.

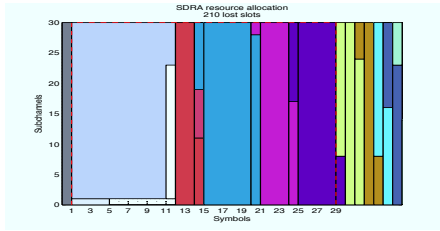
A. Simulation parameters

We focused here on a TDD WiMAX system, with an DL:UL ratio of 2:1. So the DL subframe lasts a 29-symbol duration, whose first one is preamble. We considered a 10 MHz bandwidth divided into 30 subchannels. As a result, the surface we have for burst mapping is of size 28×30, giving 840 slots.

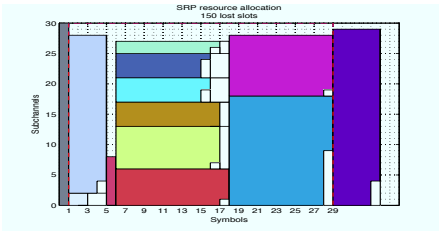
As [13] mentions, the FCH represents 24 bits. The DL-MAP has a fixed 88-bit part, plus 60 extra bits for each burst. The FCH and the DL-MAP are repeated 4 times and are transmitted with QSPK 1/2. Since we do not consider any UL subframe, only a burst of random size will act as the UL-MAP.



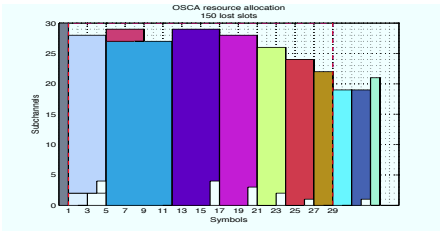
(a) Bucket-based



(b) SDRA



(c) SRP



(d) OCSA

Fig. 2. Reference algorithms sample allocation

We considered the 9 Modulation and Coding Scheme (MCS) levels of [14], from QPSK 1/2 to 64QAM. To reduce the DL-MAP size, we gathered all users of the same MCS into the same burst. As a result, there are the FCH, the DL-MAP, 9 DL bursts plus an extra burst for the UL-MAP to place on the DL-subframe. With the 0.937 bit/s/Hz spectral efficiency of QPSK 1/2 [13], FCH requires 4 slots, and a 10-burst DL-MAP requires 102 slots.

In this primary work, we wanted to focus of burst mapping efficiency, and compare our new technique with some references, without interference from different scheduling techniques. For that purpose, the size of each burst has been generated randomly, such that the sum of each burst requirement plus the FCH and a 10-burst DL-MAP correspond to the 840 available slots. As a consequence, due to ceiling operations or increased DL-MAP size due to SDRA's divided bursts, all bursts can not fit on the subframe. We will compare the number of overflowing slots for each algorithm.

B. A sample output result

First let us present a single snapshot allocation. The slots have been randomly shared between bursts as Table I details. The allocation produced by reference algorithms are presented in Fig. 2, while Fig. 3 shows our sqTM allocation. The first dark grey stripe is the preamble, and the two light grey rectangles of the lower left corner are the FCH and the DL-MAP. Reference algorithms waste a lot more slots. The bucket-based algorithm, shown in Fig. 2(a), wastes 180 slots, with in average $\frac{1+H}{2}$ wasted slots per burst. SDRA achieves to virtually waste no slot at all, but because it splits the 10 bursts into 23 rectangle allocations, the DL-MAP is more than doubled, and as Fig. 2(b) shows, 210 slots overflow from the burst allocation area. Both SRP and OCSA manage to limit the waste to 150 slots in a quite similar way. With its strategy of growing width stripes, SRP makes large stripes of multiple bursts (Fig. 2(c)), while OCSA and its narrowest stripe strategy make a lot of thin stripes (Fig. 2(d)).

FCH	DL MAP	UL-MAP	Burst #1	Burst #2	Burst #3
4	102	71	8	189	109
Burst #4	Burst #5	Burst #6	Burst #7	Burst #8	Burst #9
141	76	44	38	37	21

TABLE I
SAMPLE SNAPSHOT OF BURST SIZES IN SLOTS.

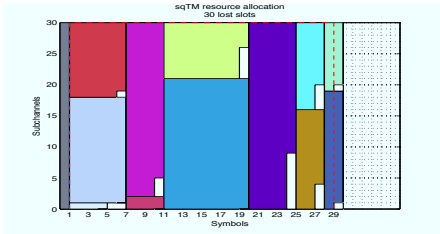


Fig. 3. sqTM sample allocation

As one can see on Fig. 3, sqTM manages to ceil bursts needs with a very limited waste. For this snapshot, all the unused slots of the burst allocation area correspond to only one overflowing column of 30 slots wasted by sqTM. This performance can be achieved thanks to our growing stripe strategy, where decisions are taken to minimize wasted slots.

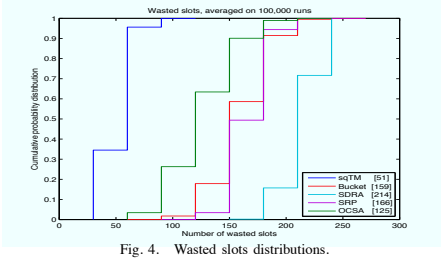


Fig. 4. Wasted slots distributions.

C. Averaged results

We also averaged the number of wasted slots by each algorithm on $100,000$ runs. The cumulative probability function and the mean of these wastes are presented in Fig. 4. Distributions are not smooth curves since wastes can only be a multiple of the number of subchannels (30 here). As one can see, sqTM manages to reduce the amount of wasted slots by 60% compared to OCSA, or even by 75% compared to SDRA.

Considering the 840 slots available for allocation, it means that in average, sqTM manages to only waste 6% of the slots, while the bucket-based algorithm, SDRA, SRP and OCSA respectively waste 19%, 25%, 20%, and 15% of the available slots, which seems quite impressive.

D. Complexity

We already demonstrated the important reduction of wastes sqTM can provide, but let us now have a look at its complexity. Even if it has a $\mathcal{O}(n^2)$ complexity like OCSA while the others three are in $\mathcal{O}(n)$, this is not a real issue since users can be gathered into a limited number of bursts, according to their channel quality. Actually, running time is more relevant. Table II presents the running time of our MATLAB scripts to perform 100,000 allocations. As one can see, OCSA in $\mathcal{O}(n^2)$ is as fast as SDRA which is in $\mathcal{O}(n)$.

As far as sqTM is concerned, it required 36.7s to perform the allocations, so it is 2 times slower than OCSA, or even 6 times slower than the bucket-based allocation. Computations are heavier and take more time with sqTM than with the other reference algorithms, but 36.7s for 100,000 allocations corresponds to $367\mu s$ per allocation. This is much shorter than the 5ms frame duration so it is not really an issue.

VI. Conclusions

In this paper, we tackled the problem of burst mapping in DL WiMAX systems, after scheduling been performed, when it comes to allocate specific subchannels for a few symbol durations. This problem is actually NP-Hard, and only a few heuristics have been proposed to solve it. Allocations must have a rectangular shape, and match both subchannel and symbol scales. Needs therefore have to be ceiled, and this leads to wastes which we have to limit.

We introduced a new burst mapping scheme named sqTM. It is based on a treemap visualization algorithm. From our numerical simulations, we obtained between 60% and 75% reduction of wasted slots compared to the reference algorithms we implemented. These limited wastes are able to provide better cell throughput and larger cell capacities from the same radio resources. As far as the running times are concerned, it appears our algorithm is significantly slower, but still fast enough to meet the time constraint of 5 ms frames.

For future work we should consider more deeply how to deal with these wasted slots and the overflow they cause. They are considered allocated by the scheduling algorithm, but actually can not be used. If bursts have to be dropped, it seems be preferable to drop the bursts of the lowest PHY-modes to maintain the highest cell throughput. The rescheduling of these bursts should be investigated.

Algorithm	Complexity	Running time
sqTM	$\mathcal{O}(n^2)$	36.7 s
Bucket	$\mathcal{O}(n)$	5.7 s
SDRA	$\mathcal{O}(n)$	22.8 s
SRP	$\mathcal{O}(n)$	27.9 s
OCSA	$\mathcal{O}(n^2)$	19.3 s

TABLE II
Running time for 100,000 allocations.

References

[1] "Draft standard for local and metropolitan area networks part 16: Air interface for broadband wireless access systems," IEEE Unapproved Draft Std P802.16 Rev2/D2, Tech. Rep., Dec. 2007.

[2] M. Maqbool, M. Coupechoux, and P. Godlewski, "Subcarrier Permutation Types in IEEE 802.16e," ENST (Télécom Paris), Tech. Rep., Apr. 2008.

[3] S. Martello and P. Toth, *Knapsack problems: Algorithms and computer implementations*. John Wiley & Sons, 1990.

[4] T. Ohseki, M. Morita, and T. Inoue, "Burst construction and packet mapping scheme for ofdma downlinks in ieee 802.16 systems," in *GLOBECOM '07*, 2007, pp. 4307 –4311.

[5] A. Erta, C. Cicconetti, and L. Lenzini, "A downlink data region allocation algorithm for ieee 802.16e ofdma," in *6th International Conf. on Information, Communications Signal Processing*, 2007, pp. 1–5.

[6] Y. Ben-Shimol, I. Kitroser, and Y. Dinitz, "Two-dimensional mapping for wireless ofdma systems," *Broadcasting, IEEE Transactions on*, vol. 52, no. 3, pp. 388–396, 2006.

[7] A. Israeli, D. Rawitz, and O. Sharon, "On the complexity of sequential rectangle placement in ieee 802.16/wimax systems," *Inf. Comput.*, vol. 206, pp. 1334–1345, November 2008.

[8] R. Cohen and L. Katzir, "Computational analysis and efficient algorithms for micro and macro ofdma scheduling," in *The 27th Conf. on Computer Comm. INFOCOM 2008.*, 2008, pp. 511–519.

[9] C. So-In, R. Jain, and A.-K. Al Tamimi, "Ocsa: An algorithm for burst mapping in ieee 802.16e mobile wimax networks," in *15th Asia-Pacific Conference on Communications APCC.*, 2009, pp. 52–58.

[10] C. So-In, R. Jain, A.-K. C. S.-I. Al Tamimi, R. Jain, A.-K. C. S.-I. Al Tamimi, R. Jain, and A.-K. Al Tamimi, "eocsa: An algorithm for burst mapping with strict qos requirements in ieee 802.16e mobile wimax networks," in *Wireless Days (WD), 2nd IFIP*, 2009, pp. 1–5.

[11] M. Bruls, K. Huizing, and J. J. van Wijk, "Squarified Treemaps," in *Proceedings of the Joint Eurographics and IEEE TCVG Symposium on Visualization VisSym'99, Vienna (Austria)*, May 1999, pp. 33–42.

[12] B. Shneiderman, "Tree visualization with tree-maps: 2-d space-filling approach," *ACM Transactions on Graphics*, vol. 11, no. 1, pp. 92–99, Jan. 1992.

[13] C. So-In, R. Jain, and A.-K. Tamimi, "Capacity Evaluation for IEEE 802.16e mobile WiMAX," *J. Comp. Sys., Netw., and Comm.*, pp. 2:1–2:1, January 2010.

[14] T. Celcer, T. Javornik, and G. Kandus, "Fairness oriented scheduling algorithm with qos support for broadband mimo systems with heterogeneous traffic," Jozef Stefan Institute, Slovenia, COST 2100 TD(09)927, Sep. 2009.

Appendix B

Uniform users distribution within an hexagonal cell

This chapter details the probability density functions for uniform distribution of users within a sector cell. A first section details the conversion of users distributions in a sector coordinate system with axis parallel to the borders of a sector into a cartesian coordinate system. Then another section details the conversion of these probability distributions into polar coordinates system.

B.1 Sector to cartesian coordinates

Figure B.1 displays a cell with cartesian coordinates (x, y) and a set of another coordinates (u, v) with axis parallel to sector of interest boundaries.

The cartesian coordinates depend on sector coordinates as follows:

$$\left\{ \begin{array}{rcrcrcl} x & = & u & \cos\frac{\pi}{3} & + v & \cos\frac{\pi}{3} & = \frac{1}{2}(u+v) \\ y & = & -u & \cos\frac{\pi}{6} & + v & \cos\frac{\pi}{6} & = \frac{\sqrt{3}}{2}(v-u) \end{array} \right. \tag{B.1}$$

with $u,\ v \in [0, R]$, $x \in [0, R]$ and $y \in \left[-\frac{\sqrt{3}}{2}R,\ \frac{\sqrt{3}}{2}R\right]$.

From Eq. (B.1), we can express the sector coordinates according to the cartesian ones:

$$\left\{ \begin{array}{rcl} u & = & x - \frac{\sqrt{3}}{3}y \\ v & = & x + \frac{\sqrt{3}}{3}y \end{array} \right. \tag{B.2}$$

Users are uniformly distributed within the sector area. That means their density function according to axis u and v are uniform:

$$T_U(u) = \left\{ \begin{array}{ll} \frac{1}{R} & \text{if } 0 \leq u \leq R \\ 0 & \text{if not} \end{array} \right. \tag{B.3}$$

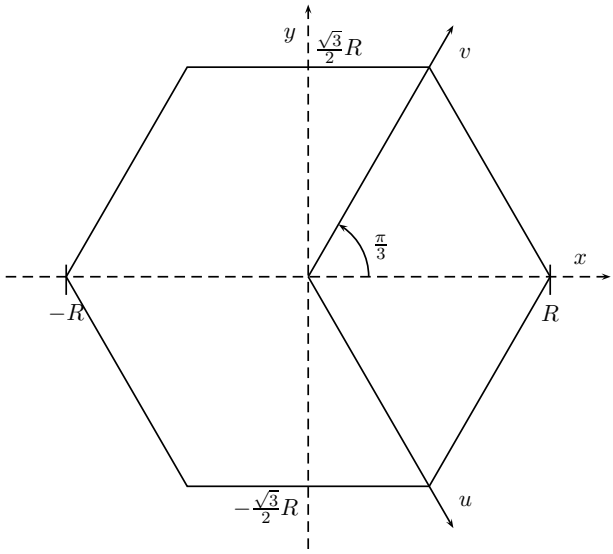


Figure B.1: Cartesian and sector coordinates in a hexagonal cell

$$T_V(v) = \begin{cases} \frac{1}{R} & \text{if } 0 \leq v \leq R \\ 0 & \text{if not} \end{cases} \tag{B.4}$$

What we are looking for is the density function according to the cartesian coordinates $T_{XY}(x, y)$, which can be obtained from

$$\begin{aligned} T_{XY}(x,y) &= \det \begin{bmatrix} \frac{\delta u}{\delta x} & \frac{\delta u}{\delta y} \\ \frac{\delta v}{\delta x} & \frac{\delta v}{\delta y} \end{bmatrix} \cdot T_{UV}(u,v) && \text{(B.5)} \\ &= \det \begin{bmatrix} 1 & \frac{-\sqrt{3}}{3} \\ 1 & \frac{\sqrt{3}}{3} \end{bmatrix} \cdot T_U(u) \cdot T_V(v) && \text{(B.6)} \\ &= \frac{2\sqrt{3}}{3} \cdot T_U\left(x - \frac{\sqrt{3}}{3}y\right) \cdot T_V\left(x + \frac{\sqrt{3}}{3}y\right) && \text{(B.7)} \end{aligned}$$

Then the density along the x and y axis can respectively be obtained from

$$T_X(x) = \int T_{XY}(x,y)dy \tag{B.8}$$

and

$$T_Y(y) = \int T_{XY}(x,y)dx \tag{B.9}$$

B.1.1 Density along the x axis

Transforming conditions from Eqs. (B.3) and (B.4) we respectively have

$$T_U\left(x-\frac{\sqrt{3}}{3}y\right)=\frac{1}{R}\quad\text{if}\quad\sqrt{3}(x-R)\le y\le\sqrt{3}x \tag{B.10}$$

and

$$T_V\left(x+\frac{\sqrt{3}}{3}y\right)=\frac{1}{R}\quad\text{if}\quad-\sqrt{3}x\le y\le\sqrt{3}(R-x) \tag{B.11}$$

Therefore we have

$$T_U\left(x-\frac{\sqrt{3}}{3}y\right)\cdot T_V\left(x+\frac{\sqrt{3}}{3}y\right)=\frac{1}{R^2}$$
$$\text{if}\ \max\left\{\sqrt{3}(x-R),-\sqrt{3}x\right\}\le y\le\min\left\{\sqrt{3}x,\sqrt{3}(R-x)\right\} \tag{B.12}$$

If $x\in\left[0,\frac{R}{2}\right]$, we have

$$\sqrt{3}(x-R)\le-\sqrt{3}x\ \Rightarrow\ \max\left\{\sqrt{3}(x-R),-\sqrt{3}x\right\}=-\sqrt{3}x \tag{B.13}$$
$$\sqrt{3}x\le\sqrt{3}(R-x)\ \Rightarrow\ \min\left\{\sqrt{3}x,\sqrt{3}(R-x)\right\}=\sqrt{3}x \tag{B.14}$$

As a result, we obtain

$$T_X(x)=\int_{-\frac{\sqrt{3}}{2}R}^{\frac{\sqrt{3}}{2}R}\frac{2\sqrt{3}}{2}\cdot T_U\left(x-\frac{\sqrt{3}}{3}y\right)\cdot T_V\left(x+\frac{\sqrt{3}}{3}y\right)dy \tag{B.15}$$
$$=\int_{-\sqrt{3}x}^{\sqrt{3}x}\frac{2\sqrt{3}}{3R^2}\,dy=\frac{2\sqrt{3}}{3R^2}\left[\ y\ \right]_{-\sqrt{3}x}^{\sqrt{3}x}=\frac{2\sqrt{3}}{3R^2}\,2\sqrt{3}x \tag{B.16}$$
$$=\frac{4x}{R^2}\quad\text{if}\quad x\in\left[0,\frac{R}{2}\right] \tag{B.17}$$

And if $x\in\left[\frac{R}{2},R\right]$, we have

$$\sqrt{3}(x-R)\ge-\sqrt{3}x\ \Rightarrow\ \max\left\{\sqrt{3}(x-R),-\sqrt{3}x\right\}=\sqrt{3}(x-R) \tag{B.18}$$
$$\sqrt{3}x\ge\sqrt{3}(R-x)\ \Rightarrow\ \min\left\{\sqrt{3}x,\sqrt{3}(R-x)\right\}=\sqrt{3}(R-x) \tag{B.19}$$

which gives us

$$T_X(x) = \int_{-\frac{\sqrt{3}}{2}R}^{\frac{\sqrt{3}}{2}R} \frac{2\sqrt{3}}{2} \cdot T_U\left(x - \frac{\sqrt{3}}{3}y\right) \cdot T_V\left(x + \frac{\sqrt{3}}{3}y\right) dy \quad \text{(B.20)}$$

$$= \int_{-\sqrt{3}(x-R)}^{\sqrt{3}(R-x)} \frac{2\sqrt{3}}{3R^2}\, dy = \frac{2\sqrt{3}}{3R^2} \Big[\, y \,\Big]_{\sqrt{3}(x-R)}^{\sqrt{3}(R-x)} \quad \text{(B.21)}$$

$$= \frac{2\sqrt{3}}{3R^2}\, 2\sqrt{3}(R-x) = \frac{4(R-x)}{R^2} \quad \text{if } x \in \left[\frac{R}{2}, R\right] \quad \text{(B.22)}$$

We finally obtain

$$T_X(x) = \begin{cases} \frac{4x}{R^2} & \text{if } x \in \left[0, \frac{R}{2}\right] \\ \frac{4(R-x)}{R^2} & \text{if } x \in \left[\frac{R}{2}, R\right] \\ 0 & \text{if not} \end{cases} \quad \text{(B.23)}$$

Verification:

$$\int_{-\infty}^{\infty} T_X(x)dx = \int_0^{\frac{R}{2}} \frac{4x}{R^2}dx + \int_{\frac{R}{2}}^{R} \frac{4(R-x)}{R^2}dx \quad \text{(B.24)}$$

$$= \frac{4}{R^2}\Big[\frac{x^2}{2}\Big]_0^{\frac{R}{2}} + \frac{4}{R^2}\Big[Rx - \frac{x^2}{2}\Big]_{\frac{R}{2}}^{R} \quad \text{(B.25)}$$

$$= \frac{4}{R^2}\, R^2 \left[\frac{1}{8} - 0 + 1 - \frac{1}{2} - \frac{1}{2} + \frac{1}{8}\right] \quad \text{(B.26)}$$

$$= 1 \;\Rightarrow\; \text{OK.} \quad \text{(B.27)}$$

B.1.2 Density along the y axis

As for the x axis, transforming conditions from Eqs. (B.3) and (B.4) we respectively have

$$T_U\left(x - \frac{\sqrt{3}}{3}y\right) = \frac{1}{R} \quad \text{if } \frac{\sqrt{3}}{3}y \le x \le R + \frac{\sqrt{3}}{3}y \quad \text{(B.28)}$$

and

$$T_V\left(x + \frac{\sqrt{3}}{3}y\right) = \frac{1}{R} \quad \text{if } \frac{-\sqrt{3}}{3}y \le x \le R - \frac{\sqrt{3}}{3}y \quad \text{(B.29)}$$

We therefore have

$$T_U\left(x - \frac{\sqrt{3}}{3}y\right) \cdot T_V\left(x + \frac{\sqrt{3}}{3}y\right) = \frac{1}{R^2}$$

$$\text{if } \max\left\{\frac{\sqrt{3}}{3}y, -\frac{\sqrt{3}}{3}y\right\} \le y \le \min\left\{R + \frac{\sqrt{3}}{3}y, R - \frac{\sqrt{3}}{3}y\right\} \quad \text{(B.30)}$$

If $y \leq 0$, we have

$$\max\left\{\frac{\sqrt{3}}{3}y, -\frac{\sqrt{3}}{3}y\right\} = -\frac{\sqrt{3}}{3}y \quad \text{(B.31)}$$

$$\min\left\{R + \frac{\sqrt{3}}{3}y, R - \frac{\sqrt{3}}{3}y\right\} = R + \frac{\sqrt{3}}{3}y \quad \text{(B.32)}$$

As a result, we obtain

$$T_Y(y) = \int_0^R \frac{2\sqrt{3}}{2} \cdot T_U\left(x - \frac{\sqrt{3}}{3}y\right) \cdot T_V\left(x + \frac{\sqrt{3}}{3}y\right) dx \quad \text{(B.33)}$$

$$= \int_{-\frac{\sqrt{3}}{3}y}^{R+\frac{\sqrt{3}}{3}y} \frac{2\sqrt{3}}{3R^2}\, dx = \frac{2\sqrt{3}}{3R^2}\left[\, y \,\right]_{-\frac{\sqrt{3}}{3}y}^{R+\frac{\sqrt{3}}{3}y} \quad \text{(B.34)}$$

$$= \frac{2\sqrt{3}}{3R^2}\left(R + \frac{2\sqrt{3}}{3}y\right) \quad \text{if } y \leq 0 \quad \text{(B.35)}$$

And if $y > 0$, we have

$$\max\left\{\frac{\sqrt{3}}{3}y, -\frac{\sqrt{3}}{3}y\right\} = \frac{\sqrt{3}}{3}y \quad \text{(B.36)}$$

$$\min\left\{R + \frac{\sqrt{3}}{3}y, R - \frac{\sqrt{3}}{3}y\right\} = R - \frac{\sqrt{3}}{3}y \quad \text{(B.37)}$$

As a result, we obtain

$$T_Y(y) = \int_0^R \frac{2\sqrt{3}}{2} \cdot T_U\left(x - \frac{\sqrt{3}}{3}y\right) \cdot T_V\left(x + \frac{\sqrt{3}}{3}y\right) dx \quad \text{(B.38)}$$

$$= \int_{\frac{\sqrt{3}}{3}y}^{R-\frac{\sqrt{3}}{3}y} \frac{2\sqrt{3}}{3R^2}\, dx = \frac{2\sqrt{3}}{3R^2}\left[\, y \,\right]_{\frac{\sqrt{3}}{3}y}^{R-\frac{\sqrt{3}}{3}y} \quad \text{(B.39)}$$

$$= \frac{2\sqrt{3}}{3R^2}\left(R - \frac{2\sqrt{3}}{3}y\right) \quad \text{if } y > 0 \quad \text{(B.40)}$$

We finally obtain

$$T_Y(y) = \begin{cases} \frac{2\sqrt{3}}{3R^2}\left(R + \frac{2\sqrt{3}}{3}y\right) & \text{if } y \in \left[-\frac{\sqrt{3}}{2}R, 0\right] \\ \frac{2\sqrt{3}}{3R^2}\left(R - \frac{2\sqrt{3}}{3}y\right) & \text{if } y \in \left[0, \frac{\sqrt{3}}{2}R\right] \\ 0 & \text{if not} \end{cases} \quad \text{(B.41)}$$

Verification:

$$\int_{-\infty}^{\infty} T_Y(y)dy \tag{B.42}$$

$$= \int_{-\frac{\sqrt{3}}{2}R}^{0} \frac{2\sqrt{3}}{3R^2}\left(R + \frac{2\sqrt{3}}{3}y\right)dy + \int_{0}^{\frac{\sqrt{3}}{2}R} \frac{2\sqrt{3}}{3R^2}\left(R - \frac{2\sqrt{3}}{3}y\right)dy \tag{B.43}$$

$$= \frac{2\sqrt{3}}{3R^2}\left(\int_{-\frac{\sqrt{3}}{2}R}^{\frac{\sqrt{3}}{2}R} R\,dy + \int_{-\frac{\sqrt{3}}{2}R}^{0} \frac{2\sqrt{3}}{3}y\,dy - \int_{0}^{\frac{\sqrt{3}}{2}R} \frac{2\sqrt{3}}{3}y\,dy\right) \tag{B.44}$$

$$= \frac{2\sqrt{3}}{3R^2}\left(\Big[R\,y\Big]_{-\frac{\sqrt{3}}{2}R}^{\frac{\sqrt{3}}{2}R} + \Big[\frac{\sqrt{3}}{3}y^2\Big]_{-\frac{\sqrt{3}}{2}R}^{0} - \Big[\frac{\sqrt{3}}{3}y^2\Big]_{0}^{\frac{\sqrt{3}}{2}R}\right) \tag{B.45}$$

$$= \frac{2\sqrt{3}}{3R^2}\left(\sqrt{3}R^2 - \frac{\sqrt{3}}{4}R^2 - \frac{\sqrt{3}}{4}R^2\right) \tag{B.46}$$

$$= \frac{2\sqrt{3}}{3R^2}\,\frac{\sqrt{3}}{2}R^2 \;=\; 1 \;\Rightarrow\; \text{OK.} \tag{B.47}$$

B.2 Cartesian to polar coordinates

Now that we have derived the density functions for the cartesian coordinates, we are going to derive density functions for polar coordinates as portrayed in Fig. B.2.

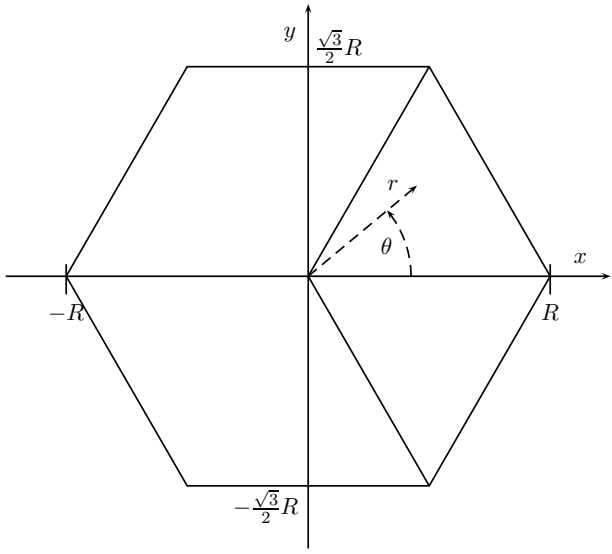


Figure B.2: Cartesian and polar coordinates in a hexagonal cell

Cartesian and polar coordinates depend on each other as follows:

$$\begin{cases} r = \sqrt{x^2 + y^2} \\ \theta = \tan^{-1}\left(\frac{y}{x}\right) \end{cases} \tag{B.48}$$

$$\begin{cases} x = r\cos(\theta) \\ y = r\sin(\theta) \end{cases} \tag{B.49}$$

with $r \in [0, R]$ and $\theta \in [-\frac{\pi}{3}, \frac{\pi}{3}]$.

Therefore the join density function $T_{R\Theta}(r, \theta)$ is

$$T_{R\Theta}(r,\theta) = \det \begin{bmatrix} \frac{\delta x}{\delta r} & \frac{\delta x}{\delta \theta} \\ \frac{\delta y}{\delta r} & \frac{\delta y}{\delta \theta} \end{bmatrix} \cdot T_{XY}(x, y) \tag{B.50}$$

$$= \det \begin{bmatrix} \cos\theta & -r\sin\theta \\ \sin\theta & r\cos\theta \end{bmatrix} \cdot \frac{2\sqrt{3}}{3} \cdot T_U\left(r\cos\theta - \frac{\sqrt{3}}{3} r\sin\theta\right) \cdot T_V\left(r\cos\theta + \frac{\sqrt{3}}{3} r\sin\theta\right) \tag{B.51}$$

$$= \frac{2\sqrt{3}}{3} r \cdot T_U\left(r\cos\theta - \frac{\sqrt{3}}{3} r\sin\theta\right) \cdot T_V\left(r\cos\theta + \frac{\sqrt{3}}{3} r\sin\theta\right) \tag{B.52}$$

B.2.1 Angular density

From Eqs. (B.3) and(B.4), we have

$$T_U\left(r\cos\theta - \frac{\sqrt{3}}{3} r\sin\theta\right) = \frac{1}{R} \quad \text{if} \quad 0 \le r\cos\theta - \frac{\sqrt{3}}{3} r\sin\theta \le R \tag{B.53}$$

$$\text{if} \quad 0 \le r \le \frac{R}{\cos\theta - \frac{\sqrt{3}}{3}\sin\theta} \tag{B.54}$$

$$T_V\left(r\cos\theta + \frac{\sqrt{3}}{3} r\sin\theta\right) = \frac{1}{R} \quad \text{if} \quad 0 \le r\cos\theta + \frac{\sqrt{3}}{3} r\sin\theta \le R \tag{B.55}$$

$$\text{if} \quad 0 \le r \le \frac{R}{\cos\theta + \frac{\sqrt{3}}{3}\sin\theta} \tag{B.56}$$

The angular density $T_\Theta(\theta)$ is obtained from

$$\begin{aligned} T_\Theta(\theta) &= \int_0^R T_{R\Theta}(r,\theta)\, dr && \text{(B.57)} \\ &= \frac{2\sqrt{3}}{3} \int_0^{\min\left\{R, \frac{R}{\cos\theta - \frac{\sqrt{3}}{3}\sin\theta}, \frac{R}{\cos\theta + \frac{\sqrt{3}}{3}\sin\theta}\right\}} \frac{r}{R^2}\, dr && \text{(B.58)} \\ &= \frac{2\sqrt{3}}{3}\, \frac{1}{R^2}\, \left[\frac{r^2}{2}\right]_0^{\min\left\{R, \frac{R}{\cos\theta - \frac{\sqrt{3}}{3}\sin\theta}, \frac{R}{\cos\theta + \frac{\sqrt{3}}{3}\sin\theta}\right\}} && \text{(B.59)} \\ &= \frac{\sqrt{3}}{3}\, \frac{1}{R}\, \left[r^2\right]_0^{\min\left\{1, \frac{1}{\cos\theta - \frac{\sqrt{3}}{3}\sin\theta}, \frac{1}{\cos\theta + \frac{\sqrt{3}}{3}\sin\theta}\right\}} && \text{(B.60)} \end{aligned}$$

And we have that

$$\min\left\{1, \frac{1}{\cos\theta - \frac{\sqrt{3}}{3}\sin\theta}, \frac{1}{\cos\theta + \frac{\sqrt{3}}{3}\sin\theta}\right\} = \begin{cases} \frac{1}{\cos\theta + \frac{\sqrt{3}}{3}\sin\theta} & \text{if} \quad \theta \geq 0 \\ \frac{1}{\cos\theta - \frac{\sqrt{3}}{3}\sin\theta} & \text{if} \quad \theta < 0 \end{cases} \qquad \text{(B.61)}$$

So finally,

$$T_\Theta(\theta) = \begin{cases} \frac{3\sqrt{3}}{\left(3\cos\theta + \sqrt{3}\sin\theta\right)^2} & \text{if} \quad \theta \in [0, \frac{\pi}{3}] \\ \frac{3\sqrt{3}}{\left(3\cos\theta - \sqrt{3}\sin\theta\right)^2} & \text{if} \quad \theta \in [-\frac{\pi}{3}, 0] \end{cases} \qquad \text{(B.62)}$$

Verification:

$$\begin{aligned} \int_{-\infty}^{\infty} T_\Theta(\theta) d\theta &= 3\sqrt{3} \int_{-\frac{\pi}{3}}^{0} \frac{d\theta}{\left(3\cos\theta - \sqrt{3}\sin\theta\right)^2} \\ &\qquad + 3\sqrt{3} \int_0^{\frac{\pi}{3}} \frac{d\theta}{\left(3\cos\theta + \sqrt{3}\sin\theta\right)^2} && \text{(B.63)} \\ &= 6\sqrt{3} \int_0^{\frac{\pi}{3}} \frac{d\theta}{\left(3\cos\theta + \sqrt{3}\sin\theta\right)^2} && \text{(B.64)} \end{aligned}$$

Or, from [85, (2.557.5) p.170], we have

$$\int \frac{dx}{(a\cos x + b\sin x)^2} = \frac{1}{a^2 + b^2} \cdot \frac{a\sin x - b\cos y}{a\cos x + b\sin x}$$

So

$$\int_{-\infty}^{\infty} T_\Theta(\theta)d\theta = 6\sqrt{3}\frac{1}{9+3}\left[\frac{3\sin\theta-\sqrt{3}\cos\theta}{3\cos\theta+\sqrt{3}\sin\theta}\right]_0^{\frac{\pi}{3}} \quad \text{(B.65)}$$

$$= \frac{\sqrt{3}}{2}\left[\frac{3\frac{\sqrt{3}}{2}-\sqrt{3}\frac{1}{2}}{3\frac{1}{2}+\sqrt{3}\frac{\sqrt{3}}{2}}-\frac{3\ 0-\sqrt{3}\ 1}{3\ 1+\sqrt{3}\ 0}\right] \quad \text{(B.66)}$$

$$= \frac{\sqrt{3}}{2}\left[\frac{\sqrt{3}}{3}+\frac{\sqrt{3}}{3}\right] = 1 \Rightarrow \text{OK.} \quad \text{(B.67)}$$

B.2.2 Radial density

The radial density $T_R(r)$ is obtained from

$$T_R(r) = \int_{-\frac{\pi}{3}}^{\frac{\pi}{3}} T_{R\Theta}(r,\theta)\ d\theta \quad \text{(B.68)}$$

$$= \frac{2\sqrt{3}}{3}r\int_{-\frac{\pi}{3}}^{\frac{\pi}{3}} T_U\left(r\cos\theta-\frac{\sqrt{3}}{3}r\sin\theta\right) \cdot\ T_V\left(r\cos\theta+\frac{\sqrt{3}}{3}r\sin\theta\right)d\theta \quad \text{(B.69)}$$

where

$$T_U\left(r\cos\theta-\frac{\sqrt{3}}{3}r\sin\theta\right) = T_U\left(r\cdot\left[\cos\theta-\frac{\sin\frac{\pi}{6}}{\cos\frac{\pi}{6}}\sin\theta\right]\right) \quad \text{(B.70)}$$

$$= T_U\left(r\cdot\frac{\cos\theta\cos\frac{\pi}{6}-\sin\theta\sin\frac{\pi}{6}}{\cos\frac{\pi}{6}}\right) \quad \text{(B.71)}$$

$$= T_U\left(\frac{r\ \cos\left(\theta+\frac{\pi}{6}\right)}{\cos\frac{\pi}{6}}\right) \quad \text{(B.72)}$$

$$= \frac{1}{R}\ \text{ if }\ 0\le\frac{r\ \cos\left(\theta+\frac{\pi}{6}\right)}{\cos\frac{\pi}{6}}\le R \quad \text{(B.73)}$$

$$= \frac{1}{R}\ \text{ if }\ 0\le\cos\left(\theta+\frac{\pi}{6}\right)\le\frac{R\cos\frac{\pi}{6}}{r} \quad \text{(B.74)}$$

and

$$T_V\left(r\cos\theta+\frac{\sqrt{3}}{3}r\sin\theta\right) = T_V\left(r\cdot\left[\cos\theta+\frac{\sin\frac{\pi}{6}}{\cos\frac{\pi}{6}}\sin\theta\right]\right) \quad \text{(B.75)}$$

$$= T_V\left(r\cdot\frac{\cos\theta\cos\frac{\pi}{6}+\sin\theta\sin\frac{\pi}{6}}{\cos\frac{\pi}{6}}\right) \quad \text{(B.76)}$$

$$= T_V\left(\frac{r\,\cos\left(\theta-\frac{\pi}{6}\right)}{\cos\frac{\pi}{6}}\right) \quad \text{(B.77)}$$

$$= \frac{1}{R} \text{ if } 0\le\frac{r\,\cos\left(\theta-\frac{\pi}{6}\right)}{\cos\frac{\pi}{6}}\le R \quad \text{(B.78)}$$

$$= \frac{1}{R} \text{ if } 0\le\cos\left(\theta-\frac{\pi}{6}\right)\le\frac{R\cos\frac{\pi}{6}}{r} \quad \text{(B.79)}$$

Since $\theta\in\left[-\frac{\pi}{3},\frac{\pi}{3}\right]$, we have

$$0\le\cos\left(\theta+\frac{\pi}{6}\right) \text{ and } 0\le\cos\left(\theta-\frac{\pi}{6}\right) \quad \text{(B.80)}$$

Now if $r<R\cos\frac{\pi}{6}=\frac{\sqrt{3}}{2}R$, all θ values are valid.

$$\frac{R\cos\frac{\pi}{6}}{r}>1 \;\Rightarrow\; \cos\left(\theta\pm\frac{\pi}{6}\right)<\frac{R\cos\frac{\pi}{6}}{r} \quad \forall\,\theta\in\left[-\frac{\pi}{3},\frac{\pi}{3}\right] \quad \text{(B.81)}$$

So

$$T_R(r) = \frac{2\sqrt{3}}{3}r\int_{-\frac{\pi}{3}}^{\frac{\pi}{3}}\frac{d\theta}{R^2} \quad \text{(B.82)}$$

$$= \frac{4\sqrt{3}}{9}\frac{r}{R^2}\pi \quad \text{if } r<\frac{\sqrt{3}}{2}R \quad \text{(B.83)}$$

But as Fig. B.3 pictures, it is only within $\left[-\frac{\pi}{3},-\theta_1\right]$, $[-\theta_2,-\theta_2]$ and $\left[\theta_1,\frac{\pi}{3}\right]$ that we can have $r\ge\frac{\sqrt{3}}{2}R$. The other θ values are not valid.

Therefore if $r\ge\frac{\sqrt{3}}{2}R$,

$$T_R(r)=\frac{2\sqrt{3}}{3}\frac{r}{R^2}\left[\int_{-\frac{\pi}{3}}^{-\theta_1}d\theta+\int_{-\theta_2}^{0}d\theta+\int_{0}^{-\theta_2}d\theta+\int_{\theta_1}^{\frac{\pi}{3}}d\theta\right] \quad \text{(B.84)}$$

where the values of θ_1 and θ_2 are derived from

$$r\ \cos\left(\theta_1 - \tfrac{\pi}{6}\right) = \tfrac{\sqrt{3}}{2}R \tag{B.85}$$

$$\Leftrightarrow \quad \theta_1 - \tfrac{\pi}{6} = \arccos\left(\tfrac{\sqrt{3}}{2}\tfrac{R}{r}\right) \tag{B.86}$$

$$\Leftrightarrow \quad \theta_1 = \tfrac{\pi}{6} + \arccos\left(\tfrac{\sqrt{3}}{2}\tfrac{R}{r}\right) \tag{B.87}$$

$$r\ \cos\left(-\theta_2 + \tfrac{\pi}{6}\right) = \tfrac{\sqrt{3}}{2}R \tag{B.88}$$

$$\Leftrightarrow \quad -\theta_2 + \tfrac{\pi}{6} = \arccos\left(\tfrac{\sqrt{3}}{2}\tfrac{R}{r}\right) \tag{B.89}$$

$$\Leftrightarrow \quad \theta_2 = \tfrac{\pi}{6} - \arccos\left(\tfrac{\sqrt{3}}{2}\tfrac{R}{r}\right) \tag{B.90}$$

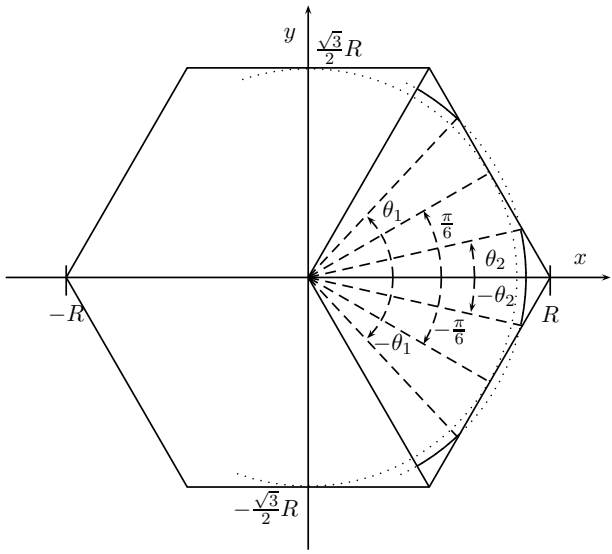


Figure B.3: Bounding angles for radial density computations

Indeed,

1. if $\theta \in \left[-\frac{\pi}{3}, -\theta_1\right]$,

$$\theta + \frac{\pi}{6} \in \left[-\frac{\pi}{6},\ -\theta_1 + \frac{\pi}{6}\right] \ \Rightarrow\ \begin{cases} \cos\left(\theta + \frac{\pi}{6}\right) \geq \cos\left(-\frac{\pi}{6}\right) \geq 0 \\ \cos\left(\theta + \frac{\pi}{6}\right) \leq \frac{\sqrt{3}}{2}\frac{R}{r} \end{cases} \tag{B.91}$$

which satisfies condition of Eq. (B.74), and

$$\theta - \frac{\pi}{6} \in \left[-\frac{\pi}{2},\ -\theta_1 - \frac{\pi}{6}\right] \Rightarrow \begin{cases} \cos\left(\theta - \frac{\pi}{6}\right) \geq \cos\left(-\frac{\pi}{2}\right) = 0 \\ \cos\left(\theta - \frac{\pi}{6}\right) \leq \cos\left(-\frac{\pi}{3} - \arccos\left(\frac{\sqrt{3}}{2}\frac{R}{r}\right)\right) \end{cases} \tag{B.92}$$

which satisfies condition of Eq. (B.79) since

$$\cos\left(-\tfrac{\pi}{3} - \arccos\left(\tfrac{\sqrt{3}}{2}\tfrac{R}{r}\right)\right) \leq \tfrac{\sqrt{3}}{2}\tfrac{R}{r} \tag{B.93}$$

$$\Leftrightarrow \quad -\tfrac{\pi}{3} - \arccos\left(\tfrac{\sqrt{3}}{2}\tfrac{R}{r}\right) \leq \arccos\left(\tfrac{\sqrt{3}}{2}\tfrac{R}{r}\right) \tag{B.94}$$

$$\Leftrightarrow \quad 2\arccos\left(\tfrac{\sqrt{3}}{2}\tfrac{R}{r}\right) \geq -\tfrac{\pi}{3} \tag{B.95}$$

which is verified since cosinus is monotonically increasing in $\left[-\frac{\pi}{2}, 0\right]$ and $R \geq r$.

2. if $\theta \in [-\theta_2, 0]$,

$$\theta + \frac{\pi}{6} \in \left[-\theta_2 + \frac{\pi}{6}, \frac{\pi}{6}\right]$$

$$\Rightarrow \quad \cos\left(\frac{\pi}{6}\right) \leq \cos\left(\theta + \frac{\pi}{6}\right) \leq \frac{\sqrt{3}}{2}\frac{R}{r} \tag{B.96}$$

which satisfies condition of Eq. (B.74), and

$$\theta - \frac{\pi}{6} \in \left[-\theta_2 - \frac{\pi}{6}, \frac{\pi}{6}\right]$$

$$\Rightarrow \quad \cos\left(-\theta_2 - \frac{\pi}{6}\right) \leq \cos\left(\theta - \frac{\pi}{6}\right) \leq \cos\left(\frac{\pi}{6}\right) \tag{B.97}$$

which satisfies condition of Eq. (B.79) since $\cos\left(\frac{\pi}{6}\right) \leq \frac{\sqrt{3}}{2}\frac{R}{r}$, because $R \geq r$.

3. if $\theta \in [0, \theta_2]$,

$$\theta + \frac{\pi}{6} \in \left[\frac{\pi}{6}, \frac{\pi}{6} + \theta_2\right]$$

$$\Rightarrow \quad \cos\left(\frac{\pi}{6}\right) \leq \cos\left(\theta + \frac{\pi}{6}\right) \leq \cos\left(\theta_2 + \frac{\pi}{6}\right) \tag{B.98}$$

which satisfies condition of Eq. (B.74) since

$$\cos\left(\tfrac{\pi}{6} + \theta_2\right) = \cos\left(\tfrac{\pi}{3} - \arccos\left(\tfrac{\sqrt{3}}{2}\tfrac{R}{r}\right)\right) \leq \tfrac{\sqrt{3}}{2}\tfrac{R}{r} \tag{B.99}$$

$$\Leftrightarrow \quad \tfrac{\pi}{3} - \arccos\left(\tfrac{\sqrt{3}}{2}\tfrac{R}{r}\right) \leq \arccos\left(\tfrac{\sqrt{3}}{2}\tfrac{R}{r}\right) \tag{B.100}$$

$$\Leftrightarrow \quad \cos\left(\tfrac{\pi}{6}\right) \leq \arccos\left(\tfrac{\sqrt{3}}{2}\tfrac{R}{r}\right) \tag{B.101}$$

which is verified since $R \geq r$, and

$$\theta - \frac{\pi}{6} \in \left[-\frac{\pi}{6}, -\frac{\pi}{6} + \theta_2\right]$$

$$\Rightarrow \quad \cos\left(\theta_2 - \frac{\pi}{6}\right) \leq \cos\left(\theta - \frac{\pi}{6}\right) \leq \cos\left(-\frac{\pi}{6}\right) \tag{B.102}$$

which satisfies condition of Eq. (B.79) since $R \geq r$.

4. if $\theta \in \left[\theta_1, \frac{\pi}{3}\right]$,

$$\theta + \frac{\pi}{6} \in \left[\theta_1 + \frac{\pi}{6}, \frac{\pi}{2}\right]$$
$$\Rightarrow \quad \cos\left(\frac{\pi}{2}\right) \leq \cos\left(\theta + \frac{\pi}{6}\right) \leq \cos\left(\theta_1 + \frac{\pi}{6}\right) \quad \text{(B.103)}$$

which satisfies condition of Eq. (B.74) since

$$\cos\left(\theta_1 + \tfrac{\pi}{6}\right) = \cos\left(\tfrac{\pi}{3} + \arccos\left(\tfrac{\sqrt{3}}{2}\tfrac{R}{r}\right)\right) \leq \tfrac{\sqrt{3}}{2}\tfrac{R}{r} \quad \text{(B.104)}$$
$$\Leftrightarrow \qquad \tfrac{\pi}{3} + \arccos\left(\tfrac{\sqrt{3}}{2}\tfrac{R}{r}\right) \geq \arccos\left(\tfrac{\sqrt{3}}{2}\tfrac{R}{r}\right) \quad \text{(B.105)}$$
$$\Leftrightarrow \qquad \tfrac{\pi}{3} \geq 0 \quad \text{(B.106)}$$

which is verified since cosinus is monotonically decreasing in $\left[0, \frac{\pi}{2}\right]$, and

$$\theta - \frac{\pi}{6} \in \left[\theta_1 - \frac{\pi}{6}, \frac{\pi}{6}\right]$$
$$\Rightarrow \quad \cos\left(\frac{\pi}{6}\right) \leq \cos\left(\theta - \frac{\pi}{6}\right) \leq \frac{\sqrt{3}}{2}\frac{R}{r} \quad \text{(B.107)}$$

which satisfies condition of Eq. (B.79).

As a result, we derive from Eq. (B.84) that for $r \geq \frac{\sqrt{3}}{2}R$,

$$T_R(r) = \frac{2\sqrt{3}}{3}\frac{r}{R^2}\left[\int_{-\frac{\pi}{3}}^{-\theta_1} d\theta + \int_{-\theta_2}^{0} d\theta + \int_{0}^{\theta_2} d\theta + \int_{\theta_1}^{\frac{\pi}{3}} d\theta\right] \quad \text{(B.108)}$$
$$= \frac{2\sqrt{3}}{3}\frac{r}{R^2}\left[-\theta_1 + \frac{\pi}{3} + 0 + \theta_2 + \theta_2 - 0 + \frac{\pi}{3} - \theta_1\right] \quad \text{(B.109)}$$
$$= \frac{4\sqrt{3}}{3}\frac{r}{R^2}\left[\frac{\pi}{3} - \theta_1 + \theta_2\right] \quad \text{(B.110)}$$
$$= \frac{8\sqrt{3}}{3}\frac{r}{R^2}\left[\frac{\pi}{6} - \arccos\left(\frac{\sqrt{3}}{2}\frac{R}{r}\right)\right] \quad \text{(B.111)}$$
$$= \frac{4\sqrt{3}}{9}\ \frac{r}{R^2}\pi - \frac{8\sqrt{3}}{3}\frac{r}{R^2}\arccos\left(\frac{\sqrt{3}}{2}\frac{R}{r}\right) \quad \text{(B.112)}$$

To summarize, we finally have that

$$T_R(r) = \begin{cases} \frac{4\sqrt{3}}{9}\ \frac{r}{R^2}\pi & \text{if } r < \frac{\sqrt{3}}{2}R \\ \frac{4\sqrt{3}}{9}\ \frac{r}{R^2}\pi - \frac{8\sqrt{3}}{3}\frac{r}{R^2}\arccos\left(\frac{\sqrt{3}}{2}\frac{R}{r}\right) & \text{if } r \geq \frac{\sqrt{3}}{2}R \end{cases} \quad \text{(B.113)}$$

Verification:

$$\int_{-\infty}^{\infty} T_R(r)dr = \underbrace{\int_0^R \frac{4\sqrt{3}}{9}\frac{r}{R^2}\pi dr}_{I_1} - \underbrace{\int_{\frac{\sqrt{3}}{2}R}^R \frac{8\sqrt{3}}{3}\frac{r}{R^2}\arccos\left(\frac{\sqrt{3}}{2}\frac{R}{r}\right)dr}_{I2} \tag{B.114}$$

where

$$\begin{aligned} I_1 &= \frac{4\sqrt{3}}{9}\frac{\pi}{R^2}\int_0^R r\ dr && \text{(B.115)}\\ &= \frac{4\sqrt{3}}{9}\frac{\pi}{R^2}\left[\frac{r^2}{2}\right]_0^R && \text{(B.116)}\\ &= \frac{2\sqrt{3}}{9}\pi && \text{(B.117)} \end{aligned}$$

and

$$I_2 = \frac{8\sqrt{3}}{3}\frac{1}{R^2}\int_{\frac{\sqrt{3}}{2}R}^R r\ \arccos\left(\frac{\sqrt{3}}{2}\frac{R}{r}\right)\ dr \tag{B.118}$$

Changing variable to $r = \frac{1}{t}$, so $dr = -\frac{1}{t^2}dt$, we obtain

$$I_2 = -\frac{8\sqrt{3}}{3}\frac{1}{R^2}\int_{\frac{2\sqrt{3}}{3}\frac{1}{R}}^{\frac{1}{R}} t^{-3}\arccos\left(\frac{t}{\frac{2\sqrt{3}}{3}R}\right)dt \tag{B.119}$$

Or, from [85, (2.832) p.238], we have

$$\int x^n \arccos\left(\frac{x}{a}\right) = \frac{x^{n+1}}{n+1}\arccos\left(\frac{x}{a}\right) + \frac{1}{n+1}\int \frac{x^{n+1}}{\sqrt{a^2-x^2}}dx$$

So

$$I_2 = -\frac{8\sqrt{3}}{3}\frac{1}{R^2}\left(\underbrace{\left[\frac{t^{-2}}{-2}\arccos\left(\frac{t}{\frac{2\sqrt{3}}{3}\frac{1}{R}}\right)\right]_{\frac{2\sqrt{3}}{3}\frac{1}{R}}^{\frac{1}{R}}}_{I3} + \frac{1}{-2}\underbrace{\int_{\frac{2\sqrt{3}}{3}\frac{1}{R}}^{\frac{1}{R}}\frac{t^{-2}}{\sqrt{\frac{4}{3R^2}-t^2}}dt}_{I4}\right) \tag{B.120}$$

$$\begin{aligned} I_3 &= -\frac{1}{2}R^2\arccos\left(\frac{1}{R}\frac{\sqrt{3}}{2}R\right) + \frac{3}{4}R^2\arccos\left(\frac{2\sqrt{3}}{3R}\frac{\sqrt{3}}{2}R\right) && \text{(B.121)}\\ &= -\frac{R^2}{2}\arccos\left(\frac{\sqrt{3}}{2}\right) + \frac{3R^2}{4}\arccos(1) && \text{(B.122)}\\ &= -\frac{\pi}{12}R^2 && \text{(B.123)} \end{aligned}$$

$$I_4 = \int_{\frac{2\sqrt{3}}{3}\frac{1}{R}}^{\frac{1}{R}} \frac{t^{-2}}{\sqrt{\frac{4}{3R^2} - t^2}} dt \tag{B.124}$$

Or, from [85, (2.269.2) p.96], we have

$$\int \frac{dx}{x^2\sqrt{a+bx+cx^2}} = -\frac{\sqrt{a+bx+cx^2}}{ax} - \frac{b}{2a}\int \frac{dx}{x\sqrt{a+bx+cx^2}}$$

So

$$I_4 = -\left[\frac{\sqrt{\frac{4}{3R^2} - t^2}}{\frac{4}{3R^2}t}\right]_{\frac{2\sqrt{3}}{3}\frac{1}{R}}^{\frac{1}{R}} - \frac{0}{\frac{8}{3R^2}}\int \frac{t^{-1}\,dt}{\sqrt{\frac{4}{3R^2} - t^2}} \tag{B.125}$$

$$= -\frac{\sqrt{\frac{4}{3R^2} - \frac{1}{R^2}}}{\frac{4}{3R^2}\frac{1}{R}} + \frac{\sqrt{\frac{4}{3R^2} - \frac{4}{3}\frac{1}{R^2}}}{\frac{4}{3R^2}\frac{2\sqrt{3}}{3}\frac{1}{R}} \tag{B.126}$$

$$= \frac{\frac{1}{R}\sqrt{\frac{1}{3}}}{\frac{4}{3R^2}\frac{1}{R}} + 0 = -\frac{\sqrt{3}}{4}R^2 \tag{B.127}$$

As a result,

$$\int_{-\infty}^{\infty} T_R(r)dr = I_1 - I_2 \tag{B.128}$$

$$= \frac{2\sqrt{3}}{9}\pi + \frac{8\sqrt{3}}{3}\frac{1}{R^2}\left(I3 - \frac{1}{2}I4\right) \tag{B.129}$$

$$= \frac{2\sqrt{3}}{9}\pi + \frac{8\sqrt{3}}{3}\frac{1}{R^2}\left(-\frac{\pi}{12}R^2 + \frac{1}{2}\frac{\sqrt{3}}{4}R^2\right) \tag{B.130}$$

$$= \frac{2\sqrt{3}}{9}\pi + \frac{2\sqrt{3}}{3}\left(-\frac{\pi}{3} + \frac{\sqrt{3}}{2}\right) \tag{B.131}$$

$$= \frac{2\sqrt{3}}{9}\pi - \frac{2\sqrt{3}}{9}\pi + \frac{2\sqrt{3}}{3}\,\frac{\sqrt{3}}{2} \tag{B.132}$$

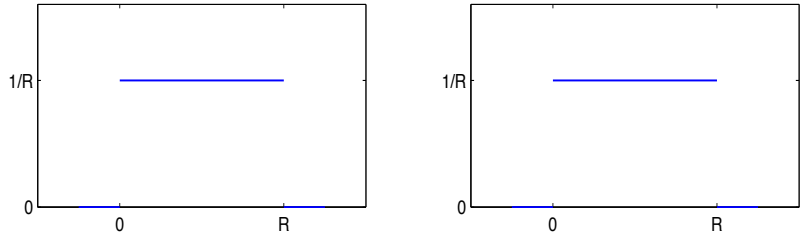
$$= 1 \Rightarrow \text{OK.} \tag{B.133}$$

B.3 Densities summary

As a result, we obtained the following densities:

$$T_U(u) = \begin{cases} \frac{1}{R} & \text{if } 0 \leq u \leq R \\ 0 & \text{if not} \end{cases} \tag{B.134}$$

$$T_V(v) = \begin{cases} \frac{1}{R} & \text{if } 0 \leq v \leq R \\ 0 & \text{if not} \end{cases} \tag{B.135}$$

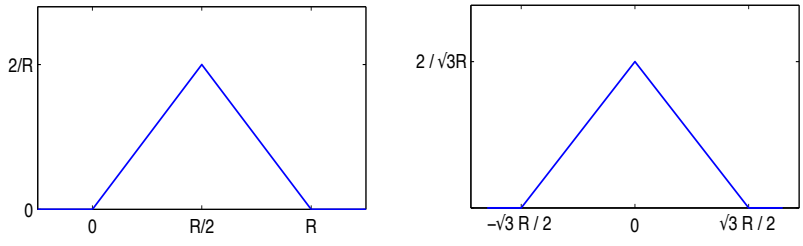


(a) $T_U(u)$ (b) $T_V(v)$

Figure B.4: Sector coordinate system densities

$$T_X(x) = \begin{cases} \frac{4x}{R^2} & \text{if } x \in \left[0, \frac{R}{2}\right] \\ \frac{4(R-x)}{R^2} & \text{if } x \in \left[\frac{R}{2}, R\right] \\ 0 & \text{if not} \end{cases} \tag{B.136}$$

$$T_Y(y) = \begin{cases} \frac{2\sqrt{3}}{3R^2}\left(R + \frac{2\sqrt{3}}{3}y\right) & \text{if } y \in \left[-\frac{\sqrt{3}}{2}R, 0\right] \\ \frac{2\sqrt{3}}{3R^2}\left(R - \frac{2\sqrt{3}}{3}y\right) & \text{if } y \in \left[0, \frac{\sqrt{3}}{2}R\right] \\ 0 & \text{if not} \end{cases} \tag{B.137}$$

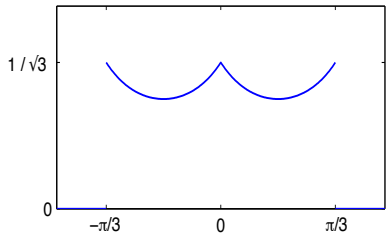


(a) $T_X(x)$ (b) $T_Y(y)$

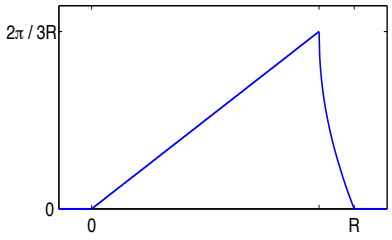
Figure B.5: Cartesian coordinate system densities

$$T_\Theta(\theta) = \begin{cases} \frac{3\sqrt{3}}{\left(3\cos\theta+\sqrt{3}\sin\theta\right)^2} & \text{if} \quad \theta \geq 0 \\ \frac{3\sqrt{3}}{\left(3\cos\theta-\sqrt{3}\sin\theta\right)^2} & \text{if} \quad \theta < 0 \end{cases} \tag{B.138}$$

$$T_R(r) = \begin{cases} \frac{4\sqrt{3}}{9}\ \frac{r}{R^2}\pi & \text{if } r < \frac{\sqrt{3}}{2}R \\ \frac{4\sqrt{3}}{9}\ \frac{r}{R^2}\pi - \frac{8\sqrt{3}}{3}\frac{r}{R^2}\arccos\left(\frac{\sqrt{3}}{2}\frac{R}{r}\right) & \text{if } r \geq \frac{\sqrt{3}}{2}R \end{cases} \tag{B.139}$$



(a) $T_\Theta(\theta)$



(b) $T_R(r)$

Figure B.6: Polar coordinate system densities

Appendix C

Probability of overlapping beams

This chapter evaluates analytically the probability of overlapping beams. First we assess the probability that two, three of four randomly steered beams do not overlap each other. Then we evaluate the probability to find, in a circular cell, two, three of four users towards whom we can steer non-overlapping beams. We finally consider users in a hexagonal cell sector, whose location is derived from our analysis of Appendix B, and evaluate the probability to find two users towards whom we can steer non-overlapping beams.

C.1 Probability that random beams are not overlapping

Considering beam directions uniformly distributed within a circular 120° sector cell, the density function $T_\Theta(\theta)$ of the beam direction is

$$T_\Theta(\theta) = \begin{cases} \frac{3}{2\pi} & \text{if } \theta \in \left[-\frac{\pi}{3},\ \frac{\pi}{3}\right] \\ 0 & \text{if not.} \end{cases} \tag{C.1}$$

$$= \frac{3}{2\pi}\ \mathbb{I}_{\left[-\frac{\pi}{3},\ \frac{\pi}{3}\right]}(\theta) \tag{C.2}$$

with $\mathbb{I}_{[a,\ b]}$ defined such as

$$\mathbb{I}_{[a,\ b]}(x) = \begin{cases} 1 & \text{if } x \in [a,\ b] \\ 0 & \text{if not.} \end{cases} \tag{C.3}$$

Considering 4 transmitting antennas, as we have mentioned in Section 4.2, the half-power beamwidth is $2\theta_B = 30° = \frac{\pi}{6}$.

C.1.1 Probability that 2 random beams are not overlapping

To evaluate the probability that 2 random beams are not overlapping, we consider a first beam randomly steered towards θ_1 and evaluate the probability that a second beam steered towards θ_2 such as $\theta_2 \notin [\theta_1 - \theta_B, \theta_1 + \theta_B\,]$.

$$
\begin{aligned}
P[\,!\,2 \text{ beams}] &= \int_{-\frac{\pi}{3}}^{\frac{\pi}{3}} \int_{-\frac{\pi}{3}}^{\frac{\pi}{3}} T_\Theta(\theta_1)\; T_\Theta(\theta_2) \\
&\quad \left(\mathbb{I}_{\left[-\frac{\pi}{3},\; \theta_1 - \theta_B\right]}(\theta_2) + \mathbb{I}_{\left[\theta_1 + \theta_B\; \frac{\pi}{3}\right]}(\theta_2) \right)\; d\theta_2 d\theta_1 \quad \text{(C.4)} \\
&= \int_{-\frac{\pi}{3}}^{\frac{\pi}{3}} T_\Theta(\theta_1) \int_{-\frac{\pi}{3}}^{\theta_1 - \theta_B} T_\Theta(\theta_2) d\theta_2\; d\theta_1 \\
&+ \int_{-\frac{\pi}{3}}^{\frac{\pi}{3}} T_\Theta(\theta_1) \int_{\theta_1 + \theta_B} \frac{\pi}{3}^{T}_{\Theta}(\theta_2) d\theta_2\; d\theta_1 \quad \text{(C.5)} \\
&= \int_{-\frac{\pi}{3}}^{\frac{\pi}{3}} T_\Theta(\theta_1)\; \frac{3}{2\pi} \left(\theta_1 - \theta_B + \frac{\pi}{3} \right)\; d\theta_1 \\
&+ \int_{-\frac{\pi}{3}}^{\frac{\pi}{3}} T_\Theta(\theta_1)\; \frac{3}{2\pi} \left(\frac{\pi}{3} - \theta_1 - \theta_B \right)\; d\theta_1 \quad \text{(C.6)} \\
&= \int_{-\frac{\pi}{3}}^{\frac{\pi}{3}} T_\Theta(\theta_1)\; \underbrace{\frac{3}{2\pi} \left(\frac{2\pi}{3} - 2\theta_B \right)}_{=1 - \frac{3\theta_B}{\pi}}\; d\theta_1 \quad \text{(C.7)} \\
&= \left(1 - \frac{3\theta_B}{\pi} \right) \underbrace{\int_{-\frac{\pi}{3}}^{\frac{\pi}{3}} T_\Theta(\theta_1)\; d\theta_1}_{=1 \text{ density function}} \quad \text{(C.8)} \\
&= \left(1 - \frac{3\theta_B}{\pi} \right) \quad \text{(C.9)}
\end{aligned}
$$

Considering 4 transmit antennas, $\theta_B = \frac{\pi}{12}$, such as

$$P[\,!\,2 \text{ beams}] = \frac{3}{4} \quad \text{(C.10)}$$

C.1.2 Probability that 3 random beams are not overlapping

$$
\begin{aligned}
P[\,!\,3 \text{ beams}] = \int_{-\frac{\pi}{3}}^{\frac{\pi}{3}} \int_{-\frac{\pi}{3}}^{\frac{\pi}{3}} \int_{-\frac{\pi}{3}}^{\frac{\pi}{3}} T_\Theta(\theta_1)\; T_\Theta(\theta_2)\; T_\Theta(\theta_3) \cdot \\
\left(\mathbb{I}_{\left[-\frac{\pi}{3},\; \theta_1-\theta_B\right]}(\theta_2) + \mathbb{I}_{\left[\theta_1+\theta_B\; \frac{\pi}{3}\right]}(\theta_2) \right) \cdot \\
\left(\mathbb{I}_{\left[-\frac{\pi}{3},\; \theta_1-\theta_B\right]}(\theta_3) + \mathbb{I}_{\left[\theta_1+\theta_B\; \frac{\pi}{3}\right]}(\theta_3) \right) \cdot \\
\left(\mathbb{I}_{\left[-\frac{\pi}{3},\; \theta_2-\theta_B\right]}(\theta_3) + \mathbb{I}_{\left[\theta_2+\theta_B\; \frac{\pi}{3}\right]}(\theta_3) \right) \cdot \\
d\theta_1\; d\theta_2\; d\theta_3 \qquad \text{(C.11)}
\end{aligned}
$$

$$
\begin{aligned}
= \int_{-\frac{\pi}{3}}^{\frac{\pi}{3}} \int_{-\frac{\pi}{3}}^{\frac{\pi}{3}} \int_{-\frac{\pi}{3}}^{\frac{\pi}{3}} T_\Theta(\theta_1)\; T_\Theta(\theta_2)\; T_\Theta(\theta_3) \cdot \\
\left(\mathbb{I}_{\left[-\frac{\pi}{3},\; \theta_1-\theta_B\right]}(\theta_2) + \mathbb{I}_{\left[\theta_1+\theta_B\; \frac{\pi}{3}\right]}(\theta_2) \right) \cdot \\
\Big(\mathbb{I}_{\left[-\frac{\pi}{3},\; \min(\theta_1,\theta_2)+\theta_B\right]}(\theta_3) + \mathbb{I}_{[\theta_2+\theta_B,\; \theta_1-\theta_B]}(\theta_3) \\
+ \mathbb{I}_{[\theta_1+\theta_B,\; \theta_2-\theta_B]}(\theta_3) + \mathbb{I}_{\left[\max(\theta_1,\theta_2)+\theta_B,\; \frac{\pi}{3}\right]}(\theta_3) \Big) \\
d\theta_1\; d\theta_2\; d\theta_3 \qquad \text{(C.12)}
\end{aligned}
$$

$$
\begin{aligned}
= \int_{-\frac{\pi}{3}}^{\frac{\pi}{3}} T_\Theta(\theta_1) \int_{-\frac{\pi}{3}}^{\theta_1-\theta_B} T_\Theta(\theta_2) \int T_\Theta(\theta_3) \\
\cdot\; \mathbb{I}_{\left[-\frac{\pi}{3},\theta_2-\theta_B\right]\cup[\theta_2+\theta_B,\theta_1-\theta_B]\cup\left[\theta_1+\theta_B,\frac{\pi}{3}\right]}(\theta_3)\; d\theta_3\; d\theta_2\; d\theta_1 \\
+ \int_{-\frac{\pi}{3}}^{\frac{\pi}{3}} T_\Theta(\theta_1) \int_{\theta_1+\theta_B}^{\frac{\pi}{3}} T_\Theta(\theta_2) \int T_\Theta(\theta_3) \\
\cdot\; \mathbb{I}_{\left[-\frac{\pi}{3},\theta_1-\theta_B\right]\cup[\theta_1+\theta_B,\theta_2-\theta_B]\cup\left[\theta_2+\theta_B,\frac{\pi}{3}\right]}(\theta_3)\; d\theta_3\; d\theta_2\; d\theta_1 \qquad \text{(C.13)}
\end{aligned}
$$

since

$$
\theta_2 \in \left[-\frac{\pi}{3}, \theta_1 - \theta_B\right] \;\Rightarrow\; -\frac{\pi}{3} \le \theta_2 \le \theta_1 - \theta_B \le \theta_1 \qquad \text{(C.14)}
$$

$$
\theta_2 \in \left[\theta_1 + \theta_B, \frac{\pi}{3}\right] \;\Rightarrow\; \theta_1 \le \theta_1 + \theta_B \le \theta_2 \le \frac{\pi}{3} \qquad \text{(C.15)}
$$

So $P[\,!\,3 \text{ beams}]$

$$= \int_{-\frac{\pi}{3}}^{\frac{\pi}{3}} T_\Theta(\theta_1) \int_{-\frac{\pi}{3}}^{\theta_1-\theta_B} T_\Theta(\theta_2)\ \frac{3}{2\pi}\left[\frac{2\pi}{3} - 4\theta_B\right]\ d\theta_2\ d\theta_1$$

$$+ \int_{-\frac{\pi}{3}}^{\frac{\pi}{3}} T_\Theta(\theta_1) \int_{\theta_1+\theta_B}^{\frac{\pi}{3}} T_\Theta(\theta_2)\ \underbrace{\frac{3}{2\pi}\left[\frac{2\pi}{3} - 4\theta_B\right]}_{=1-\frac{6\theta_B}{\pi}}\ d\theta_2\ d\theta_1 \tag{C.16}$$

$$= \left(1 - \frac{6\theta_B}{\pi}\right) \int_{-\frac{\pi}{3}}^{\frac{\pi}{3}} T_\Theta(\theta_1)\ \frac{3}{2\pi}\left(\theta_1 - \theta_B + \frac{\pi}{3}\right)\ d\theta_1$$

$$+ \left(1 - \frac{6\theta_B}{\pi}\right) \int_{-\frac{\pi}{3}}^{\frac{\pi}{3}} T_\Theta(\theta_1)\ \frac{3}{2\pi}\left(-\theta_1 - \theta_B + \frac{\pi}{3}\right)\ d\theta_1 \tag{C.17}$$

$$= \left(1 - \frac{6\theta_B}{\pi}\right) \int_{-\frac{\pi}{3}}^{\frac{\pi}{3}} T_\Theta(\theta_1)\ \underbrace{\frac{3}{2\pi}\left(-\theta_B + \frac{\pi}{3}\right)}_{=1-\frac{3\theta_B}{\pi}}\ d\theta_1 \tag{C.18}$$

$$= \left(1 - \frac{6\theta_B}{\pi}\right)\left(1 - \frac{3\theta_B}{\pi}\right)\ \frac{3}{2\pi}\ \underbrace{\left[\theta_1\right]_{-\frac{\pi}{3}}^{\frac{\pi}{3}}}_{=\frac{2\pi}{3}} \tag{C.19}$$

$$= \left(1 - \frac{6\theta_B}{\pi}\right)\left(1 - \frac{3\theta_B}{\pi}\right) \tag{C.20}$$

Considering 4 transmit antennas, $\theta_B = \frac{\pi}{12}$, such as

$$P[\,!\,3 \text{ beams}] = \left(1 - \frac{1}{2}\right)\left(1 - \frac{1}{4}\right) = \frac{3}{8} \tag{C.21}$$

C.1.3 Probability that 4 random beams are not overlapping

$$P[\,!\,4 \text{ beams}] = \int_{-\frac{\pi}{3}}^{\frac{\pi}{3}} \int_{-\frac{\pi}{3}}^{\frac{\pi}{3}} \int_{-\frac{\pi}{3}}^{\frac{\pi}{3}} \int_{-\frac{\pi}{3}}^{\frac{\pi}{3}} T_\Theta(\theta_1)\ T_\Theta(\theta_2)\ T_\Theta(\theta_3)\ T_\Theta(\theta_4)\ \cdot$$
$$\left(\mathbb{I}_{\left[-\frac{\pi}{3},\theta_1-\theta_B\right]}(\theta_2) + \mathbb{I}_{\left[\theta_1+\theta_B,\frac{\pi}{3}\right]}(\theta_2)\right)\ \cdot\ \left(\mathbb{I}_{\left[-\frac{\pi}{3},\min(\theta_1,\theta_2)-\theta_B\right]}(\theta_3) +\right.$$
$$\left.\mathbb{I}_{[\min(\theta_1,\theta_2)+\theta_B,\max(\theta_1,\theta_2)-\theta_B]}(\theta_3) + \mathbb{I}_{\left[\max(\theta_1,\theta_2)+\theta_B,\frac{\pi}{3}\right]}(\theta_3)\right)\ \cdot$$
$$\left(\mathbb{I}_{\left[-\frac{\pi}{3},\theta_1-\theta_B\right]}(\theta_4) + \mathbb{I}_{\left[\theta_1+\theta_B,\frac{\pi}{3}\right]}(\theta_4)\right)\ \cdot\ \left(\mathbb{I}_{\left[-\frac{\pi}{3},\min(\theta_2,\theta_3)-\theta_B\right]}(\theta_4) +\right.$$
$$\left.\mathbb{I}_{[\min(\theta_2,\theta_3)+\theta_B,\max(\theta_2,\theta_3)-\theta_B]}(\theta_4) + \mathbb{I}_{\left[\max(\theta_2,\theta_3)+\theta_B,\frac{\pi}{3}\right]}(\theta_4)\right)\ \cdot$$
$$d\theta_1\ d\theta_2\ d\theta_3\ d\theta_4 \tag{C.22}$$

$$
\begin{aligned}
P[\,!\,4 \text{ beams}] = \\
&\int_{-\frac{\pi}{3}}^{\frac{\pi}{3}} \int_{-\frac{\pi}{3}}^{\theta_1-\theta_B} \int_{-\frac{\pi}{3}}^{\frac{\pi}{3}} \int_{-\frac{\pi}{3}}^{\frac{\pi}{3}} T_\Theta(\theta_1)\; T_\Theta(\theta_2)\; T_\Theta(\theta_3)\; T_\Theta(\theta_4)\; \cdot \\
&\Big(\mathbb{I}_{\left[-\frac{\pi}{3},\min(\theta_1,\theta_2)-\theta_B\right]}(\theta_3) + \mathbb{I}_{[\min(\theta_1,\theta_2)+\theta_B,\max(\theta_1,\theta_2)-\theta_B]}(\theta_3) + \\
&\mathbb{I}_{\left[\max(\theta_1,\theta_2)\;+\theta_B,\frac{\pi}{3}\right]}(\theta_3)\Big)\; \cdot\; \Big(\mathbb{I}_{\left[-\frac{\pi}{3},\theta_1-\theta_B\right]}(\theta_4) + \mathbb{I}_{\left[\theta_1+\theta_B,\frac{\pi}{3}\right]}(\theta_4)\Big)\; \cdot \\
&\Big(\mathbb{I}_{\left[-\frac{\pi}{3},\min(\theta_2,\theta_3)-\theta_B\right]}(\theta_4) + \mathbb{I}_{[\min(\theta_2,\theta_3)+\theta_B,\max(\theta_2,\theta_3)-\theta_B]}(\theta_4) + \\
&\qquad \mathbb{I}_{\left[\max(\theta_2,\theta_3)+\theta_B,\frac{\pi}{3}\right]}(\theta_4)\Big)\; \cdot\; d\theta_1\; d\theta_2\; d\theta_3\; d\theta_4 \\
&+ \int_{-\frac{\pi}{3}}^{\frac{\pi}{3}} \int_{\theta_1+\theta_B}^{\frac{\pi}{3}} \int_{-\frac{\pi}{3}}^{\frac{\pi}{3}} \int_{-\frac{\pi}{3}}^{\frac{\pi}{3}} T_\Theta(\theta_1)\; T_\Theta(\theta_2)\; T_\Theta(\theta_3)\; T_\Theta(\theta_4)\; \cdot \\
&\Big(\mathbb{I}_{\left[-\frac{\pi}{3},\min(\theta_1,\theta_2)-\theta_B\right]}(\theta_3) + \mathbb{I}_{[\min(\theta_1,\theta_2)+\theta_B,\max(\theta_1,\theta_2)-\theta_B]}(\theta_3) + \\
&\mathbb{I}_{\left[\max(\theta_1,\theta_2)+\theta_B,\frac{\pi}{3}\right]}(\theta_3)\Big)\; \cdot \Big(\mathbb{I}_{\left[-\frac{\pi}{3},\theta_1-\theta_B\right]}(\theta_4) + \mathbb{I}_{\left[\theta_1+\theta_B,\frac{\pi}{3}\right]}(\theta_4)\Big)\; \cdot \\
&\Big(\mathbb{I}_{\left[-\frac{\pi}{3},\min(\theta_2,\theta_3)-\theta_B\right]}(\theta_4) + \mathbb{I}_{[\min(\theta_2,\theta_3)+\theta_B,\max(\theta_2,\theta_3)-\theta_B]}(\theta_4) + \\
&\qquad \mathbb{I}_{\left[\max(\theta_2,\theta_3)+\theta_B,\frac{\pi}{3}\right]}(\theta_4)\Big)\; \cdot\; d\theta_1\; d\theta_2\; d\theta_3\; d\theta_4 \qquad \text{(C.23)} \\
&\qquad\qquad = I_1 \;+\; I_2 \qquad \text{(C.24)}
\end{aligned}
$$

Within I_1, $\theta_2 \in \left[-\frac{\pi}{3},\, \theta_1 - \theta_B\right] \;\Rightarrow\; -\frac{\pi}{3} \le \theta_2 \le \theta_1 - \theta_B \le \theta_1$, so

$$
\begin{aligned}
I_1 \;=\; &\int_{-\frac{\pi}{3}}^{\frac{\pi}{3}} \int_{-\frac{\pi}{3}}^{\theta_1-\theta_B} \int_{-\frac{\pi}{3}}^{\frac{\pi}{3}} \int_{-\frac{\pi}{3}}^{\frac{\pi}{3}} T_\Theta(\theta_1)\; T_\Theta(\theta_2)\; T_\Theta(\theta_3)\; T_\Theta(\theta_4)\; \cdot \\
&\Big(\mathbb{I}_{\left[-\frac{\pi}{3},\theta_2-\theta_B\right]}(\theta_3) + \mathbb{I}_{[\theta_2+\theta_B,\theta_1-\theta_B]}(\theta_3) + \mathbb{I}_{\left[\theta_1+\theta_B,\frac{\pi}{3}\right]}(\theta_3)\Big)\; \cdot \\
&\Big(\mathbb{I}_{\left[-\frac{\pi}{3},\theta_1-\theta_B\right]}(\theta_4) + \mathbb{I}_{\left[\theta_1+\theta_B,\frac{\pi}{3}\right]}(\theta_4)\Big)\; \cdot\; \Big(\mathbb{I}_{\left[-\frac{\pi}{3},\min(\theta_2,\theta_3)-\theta_B\right]}(\theta_4) + \\
&\mathbb{I}_{[\min(\theta_2,\theta_3)+\theta_B,\max(\theta_2,\theta_3)-\theta_B]}(\theta_4) + \mathbb{I}_{\left[\max(\theta_2,\theta_3)+\theta_B,\frac{\pi}{3}\right]}(\theta_4)\Big)\; \cdot \\
&d\theta_1\; d\theta_2\; d\theta_3\; d\theta_4 \qquad \text{(C.25)}
\end{aligned}
$$

$$
\begin{aligned}
I_1 \;=\; & \int_{-\frac{\pi}{3}}^{\frac{\pi}{3}} \int_{-\frac{\pi}{3}}^{\theta_1-\theta_B} \int_{-\frac{\pi}{3}}^{\theta_2-\theta_B} \int_{-\frac{\pi}{3}}^{\frac{\pi}{3}} T_\Theta(\theta_1)\; T_\Theta(\theta_2)\; T_\Theta(\theta_3)\; T_\Theta(\theta_4) \;\cdot \\
& \Big(\mathbb{I}_{\left[-\frac{\pi}{3},\theta_1-\theta_B\right]}(\theta_4) + \mathbb{I}_{\left[\theta_1+\theta_B,\frac{\pi}{3}\right]}(\theta_4) \Big) \;\cdot \\
& \Big(\mathbb{I}_{\left[-\frac{\pi}{3},\theta_3-\theta_B\right]}(\theta_4) + \mathbb{I}_{[\theta_3+\theta_B,\theta_2-\theta_B]}(\theta_4) + \mathbb{I}_{\left[\theta_2+\theta_B,\frac{\pi}{3}\right]}(\theta_4) \Big) \;\cdot \\
& d\theta_1\; d\theta_2\; d\theta_3\; d\theta_4 \\
& + \int_{-\frac{\pi}{3}}^{\frac{\pi}{3}} \int_{-\frac{\pi}{3}}^{\theta_1-\theta_B} \int_{\theta_2+\theta_B}^{\theta_1-\theta_B} \int_{-\frac{\pi}{3}}^{\frac{\pi}{3}} T_\Theta(\theta_1)\; T_\Theta(\theta_2)\; T_\Theta(\theta_3)\; T_\Theta(\theta_4) \;\cdot \\
& \Big(\mathbb{I}_{\left[-\frac{\pi}{3},\theta_1-\theta_B\right]}(\theta_4) + \mathbb{I}_{\left[\theta_1+\theta_B,\frac{\pi}{3}\right]}(\theta_4) \Big) \;\cdot \\
& \Big(\mathbb{I}_{\left[-\frac{\pi}{3},\theta_2-\theta_B\right]}(\theta_4) + \mathbb{I}_{[\theta_2+\theta_B,\theta_3-\theta_B]}(\theta_4) + \mathbb{I}_{\left[\theta_3+\theta_B,\frac{\pi}{3}\right]}(\theta_4) \Big) \;\cdot \\
& d\theta_1\; d\theta_2\; d\theta_3\; d\theta_4 \\
& + \int_{-\frac{\pi}{3}}^{\frac{\pi}{3}} \int_{-\frac{\pi}{3}}^{\theta_1-\theta_B} \int_{\theta_1+\theta_B}^{\frac{\pi}{3}} \int_{-\frac{\pi}{3}}^{\frac{\pi}{3}} T_\Theta(\theta_1)\; T_\Theta(\theta_2)\; T_\Theta(\theta_3)\; T_\Theta(\theta_4) \;\cdot \\
& \Big(\mathbb{I}_{\left[-\frac{\pi}{3},\theta_1-\theta_B\right]}(\theta_4) + \mathbb{I}_{\left[\theta_1+\theta_B,\frac{\pi}{3}\right]}(\theta_4) \Big) \;\cdot \\
& \Big(\mathbb{I}_{\left[-\frac{\pi}{3},\theta_2-\theta_B\right]}(\theta_4) + \mathbb{I}_{[\theta_2+\theta_B,\theta_3-\theta_B]}(\theta_4) + \mathbb{I}_{\left[\theta_3+\theta_B,\frac{\pi}{3}\right]}(\theta_4) \Big) \;\cdot \\
& d\theta_1\; d\theta_2\; d\theta_3\; d\theta_4 \qquad \text{(C.26)}
\end{aligned}
$$

$$
\begin{aligned}
= \; & \int_{-\frac{\pi}{3}}^{\frac{\pi}{3}} \int_{-\frac{\pi}{3}}^{\theta_1-\theta_B} \int_{-\frac{\pi}{3}}^{\theta_2-\theta_B} \int_{-\frac{\pi}{3}}^{\frac{\pi}{3}} T_\Theta(\theta_1)\; T_\Theta(\theta_2)\; T_\Theta(\theta_3)\; T_\Theta(\theta_4) \;\cdot \\
& \Big(\mathbb{I}_{\left[-\frac{\pi}{3},\theta_3-\theta_B\right]}(\theta_4) + \mathbb{I}_{[\theta_3+\theta_B,\theta_2-\theta_B]}(\theta_4) + \mathbb{I}_{[\theta_2+\theta_B,\theta_1-\theta_B]}(\theta_4) \\
& \quad + \mathbb{I}_{\left[\theta_1+\theta_B,\frac{\pi}{3}\right]}(\theta_4) \Big) \quad d\theta_1\; d\theta_2\; d\theta_3\; d\theta_4 \\
& + \int_{-\frac{\pi}{3}}^{\frac{\pi}{3}} \int_{-\frac{\pi}{3}}^{\theta_1-\theta_B} \int_{\theta_2+\theta_B}^{\theta_1-\theta_B} \int_{-\frac{\pi}{3}}^{\frac{\pi}{3}} T_\Theta(\theta_1)\; T_\Theta(\theta_2)\; T_\Theta(\theta_3)\; T_\Theta(\theta_4) \;\cdot \\
& \Big(\mathbb{I}_{\left[-\frac{\pi}{3},\theta_2-\theta_B\right]}(\theta_4) + \mathbb{I}_{[\theta_2+\theta_B,\theta_3-\theta_B]}(\theta_4) + \mathbb{I}_{[\theta_3+\theta_B,\theta_1-\theta_B]}(\theta_4) \\
& \quad + \mathbb{I}_{\left[\theta_1+\theta_B,\frac{\pi}{3}\right]}(\theta_4) \Big) \quad d\theta_1\; d\theta_2\; d\theta_3\; d\theta_4 \\
& + \int_{-\frac{\pi}{3}}^{\frac{\pi}{3}} \int_{-\frac{\pi}{3}}^{\theta_1-\theta_B} \int_{\theta_1+\theta_B}^{\frac{\pi}{3}} \int_{-\frac{\pi}{3}}^{\frac{\pi}{3}} T_\Theta(\theta_1)\; T_\Theta(\theta_2)\; T_\Theta(\theta_3)\; T_\Theta(\theta_4) \;\cdot \\
& \Big(\mathbb{I}_{\left[-\frac{\pi}{3},\theta_2-\theta_B\right]}(\theta_4) + \mathbb{I}_{[\theta_2+\theta_B,\theta_1-\theta_B]}(\theta_4) + \mathbb{I}_{[\theta_1+\theta_B,\theta_3-\theta_B]}(\theta_4) \\
& \quad + \mathbb{I}_{\left[\theta_3+\theta_B,\frac{\pi}{3}\right]}(\theta_4) \Big) \quad d\theta_1\; d\theta_2\; d\theta_3\; d\theta_4 \qquad \text{(C.27)}
\end{aligned}
$$

$$
\begin{aligned}
I_1 \;=\; & \int_{-\frac{\pi}{3}}^{\frac{\pi}{3}} \int_{-\frac{\pi}{3}}^{\theta_1-\theta_B} \int_{-\frac{\pi}{3}}^{\theta_2-\theta_B} T_\Theta(\theta_1)\; T_\Theta(\theta_2)\; T_\Theta(\theta_3)\;\cdot \\
& \qquad\qquad \frac{3}{2\pi}\left[\frac{2\pi}{3}-6\theta_B\right] d\theta_1\; d\theta_2\; d\theta_3 \\
& + \int_{-\frac{\pi}{3}}^{\frac{\pi}{3}} \int_{-\frac{\pi}{3}}^{\theta_1-\theta_B} \int_{\theta_2+\theta_B}^{\theta_1-\theta_B} T_\Theta(\theta_1)\; T_\Theta(\theta_2)\; T_\Theta(\theta_3)\;\cdot \\
& \qquad\qquad \frac{3}{2\pi}\left[\frac{2\pi}{3}-6\theta_B\right] d\theta_1\; d\theta_2\; d\theta_3 \\
& + \int_{-\frac{\pi}{3}}^{\frac{\pi}{3}} \int_{-\frac{\pi}{3}}^{\theta_1-\theta_B} \int_{\theta_1-\theta_B}^{\frac{\pi}{3}} T_\Theta(\theta_1)\; T_\Theta(\theta_2)\; T_\Theta(\theta_3)\;\cdot \\
& \qquad\qquad \underbrace{\frac{3}{2\pi}\left[\frac{2\pi}{3}-6\theta_B\right]}_{=1-\frac{9\theta_B}{\pi}} d\theta_1\; d\theta_2\; d\theta_3 \qquad \text{(C.28)}
\end{aligned}
$$

$$
\begin{aligned}
= \; & \left(1-\frac{9\theta_B}{\pi}\right) \int_{-\frac{\pi}{3}}^{\frac{\pi}{3}} \int_{-\frac{\pi}{3}}^{\theta_1-\theta_B} T_\Theta(\theta_1)\; T_\Theta(\theta_2)\;\cdot \\
& \qquad\qquad \underbrace{\frac{3}{2\pi}\left[\frac{2\pi}{3}-4\theta_B\right]}_{=1-\frac{6\theta_B}{\pi}} d\theta_1\; d\theta_2 \qquad \text{(C.29)}
\end{aligned}
$$

$$
= \left(1-\frac{9\theta_B}{\pi}\right)\left(1-\frac{6\theta_B}{\pi}\right) \int_{-\frac{\pi}{3}}^{\frac{\pi}{3}} T_\Theta(\theta_1)\, \underbrace{\frac{3}{2\pi}\left[\theta_1-\theta_B+\frac{\pi}{3}\right]}_{=\frac{1}{2}\left(1-\frac{3\theta_B}{\pi}\right)} d\theta_1 \qquad \text{(C.30)}
$$

$$
= \frac{1}{2}\left(1-\frac{9\theta_B}{\pi}\right)\left(1-\frac{6\theta_B}{\pi}\right)\left(1-\frac{3\theta_B}{\pi}\right) \qquad \text{(C.31)}
$$

And within I_2, $\theta_2 \in \left[\theta_1+\theta_B,\, \frac{\pi}{3}\right] \;\Rightarrow\; \theta_1 \le \theta_1+\theta_B \le \theta_2 \le \frac{\pi}{3}$, so

$$
\begin{aligned}
I_2 \;=\; & \int_{-\frac{\pi}{3}}^{\frac{\pi}{3}} \int_{\theta_1+\theta_B}^{\frac{\pi}{3}} \int_{-\frac{\pi}{3}}^{\frac{\pi}{3}} \int_{-\frac{\pi}{3}}^{\frac{\pi}{3}} T_\Theta(\theta_1)\; T_\Theta(\theta_2)\; T_\Theta(\theta_3)\; T_\Theta(\theta_4)\;\cdot \\
& \left(\mathbb{I}_{\left[-\frac{\pi}{3},\theta_1-\theta_B\right]}(\theta_3)+\mathbb{I}_{[\theta_1+\theta_B,\theta_2-\theta_B]}(\theta_3)+\mathbb{I}_{\left[\theta_2+\theta_B,\frac{\pi}{3}\right]}(\theta_3)\right)\;\cdot \\
& \left(\mathbb{I}_{\left[-\frac{\pi}{3},\theta_1-\theta_B\right]}(\theta_4)+\mathbb{I}_{\left[\theta_1+\theta_B,\frac{\pi}{3}\right]}(\theta_4)\right)\;\cdot\; \left(\mathbb{I}_{\left[-\frac{\pi}{3},\min(\theta_2,\theta_3)-\theta_B\right]}(\theta_4)+\right. \\
& \left.\mathbb{I}_{[\min(\theta_2,\theta_3)+\theta_B,\max(\theta_2,\theta_3)-\theta_B]}(\theta_4)+\mathbb{I}_{\left[\max(\theta_2,\theta_3)+\theta_B,\frac{\pi}{3}\right]}(\theta_4)\right)\;\cdot \\
& d\theta_1\; d\theta_2\; d\theta_3\; d\theta_4 \qquad \text{(C.32)}
\end{aligned}
$$

$$
\begin{aligned}
I_2 \;=\; & \int_{-\frac{\pi}{3}}^{\frac{\pi}{3}} \int_{\theta_1+\theta_B}^{\frac{\pi}{3}} \int_{-\frac{\pi}{3}}^{\theta_1-\theta_B} \int_{-\frac{\pi}{3}}^{\frac{\pi}{3}} T_\Theta(\theta_1)\; T_\Theta(\theta_2)\; T_\Theta(\theta_3)\; T_\Theta(\theta_4) \;\cdot \\
& \Big(\mathbb{I}_{\left[-\frac{\pi}{3},\theta_1-\theta_B\right]}(\theta_4) + \mathbb{I}_{\left[\theta_1+\theta_B,\frac{\pi}{3}\right]}(\theta_4) \Big) \;\cdot \\
& \Big(\mathbb{I}_{\left[-\frac{\pi}{3},\theta_3-\theta_B\right]}(\theta_4) + \mathbb{I}_{[\theta_3+\theta_B,\theta_2-\theta_B]}(\theta_4) + \mathbb{I}_{\left[\theta_2+\theta_B,\frac{\pi}{3}\right]}(\theta_4) \Big) \;\cdot \\
& d\theta_1\; d\theta_2\; d\theta_3\; d\theta_4 \\
& + \int_{-\frac{\pi}{3}}^{\frac{\pi}{3}} \int_{\theta_1+\theta_B}^{\frac{\pi}{3}} \int_{\theta_1+\theta_B}^{\theta_2-\theta_B} \int_{-\frac{\pi}{3}}^{\frac{\pi}{3}} T_\Theta(\theta_1)\; T_\Theta(\theta_2)\; T_\Theta(\theta_3)\; T_\Theta(\theta_4) \;\cdot \\
& \Big(\mathbb{I}_{\left[-\frac{\pi}{3},\theta_1-\theta_B\right]}(\theta_4) + \mathbb{I}_{\left[\theta_1+\theta_B,\frac{\pi}{3}\right]}(\theta_4) \Big) \;\cdot \\
& \Big(\mathbb{I}_{\left[-\frac{\pi}{3},\theta_3-\theta_B\right]}(\theta_4) + \mathbb{I}_{[\theta_3+\theta_B,\theta_2-\theta_B]}(\theta_4) + \mathbb{I}_{\left[\theta_2+\theta_B,\frac{\pi}{3}\right]}(\theta_4) \Big) \;\cdot \\
& d\theta_1\; d\theta_2\; d\theta_3\; d\theta_4 \\
& + \int_{-\frac{\pi}{3}}^{\frac{\pi}{3}} \int_{\theta_1+\theta_B}^{\frac{\pi}{3}} \int_{\theta_2+\theta_B}^{\frac{\pi}{3}} \int_{-\frac{\pi}{3}}^{\frac{\pi}{3}} T_\Theta(\theta_1)\; T_\Theta(\theta_2)\; T_\Theta(\theta_3)\; T_\Theta(\theta_4) \;\cdot \\
& \Big(\mathbb{I}_{\left[-\frac{\pi}{3},\theta_1-\theta_B\right]}(\theta_4) + \mathbb{I}_{\left[\theta_1+\theta_B,\frac{\pi}{3}\right]}(\theta_4) \Big) \;\cdot \\
& \Big(\mathbb{I}_{\left[-\frac{\pi}{3},\theta_2-\theta_B\right]}(\theta_4) + \mathbb{I}_{[\theta_2+\theta_B,\theta_3-\theta_B]}(\theta_4) + \mathbb{I}_{\left[\theta_3+\theta_B,\frac{\pi}{3}\right]}(\theta_4) \Big) \;\cdot \\
& d\theta_1\; d\theta_2\; d\theta_3\; d\theta_4 \qquad \text{(C.33)}
\end{aligned}
$$

$$
\begin{aligned}
= \; & \int_{-\frac{\pi}{3}}^{\frac{\pi}{3}} \int_{\theta_1+\theta_B}^{\frac{\pi}{3}} \int_{-\frac{\pi}{3}}^{\theta_1-\theta_B} \int_{-\frac{\pi}{3}}^{\frac{\pi}{3}} T_\Theta(\theta_1)\; T_\Theta(\theta_2)\; T_\Theta(\theta_3)\; T_\Theta(\theta_4) \;\cdot \\
& \Big(\mathbb{I}_{\left[-\frac{\pi}{3},\theta_3-\theta_B\right]}(\theta_4) + \mathbb{I}_{[\theta_3+\theta_B,\theta_1-\theta_B]}(\theta_4) + \mathbb{I}_{[\theta_1+\theta_B,\theta_2-\theta_B]}(\theta_4) \\
& \quad + \mathbb{I}_{\left[\theta_2+\theta_B,\frac{\pi}{3}\right]}(\theta_4) \Big) \;\cdot\; d\theta_1\; d\theta_2\; d\theta_3\; d\theta_4 \\
& + \int_{-\frac{\pi}{3}}^{\frac{\pi}{3}} \int_{\theta_1+\theta_B}^{\frac{\pi}{3}} \int_{\theta_1+\theta_B}^{\theta_2-\theta_B} \int_{-\frac{\pi}{3}}^{\frac{\pi}{3}} T_\Theta(\theta_1)\; T_\Theta(\theta_2)\; T_\Theta(\theta_3)\; T_\Theta(\theta_4) \;\cdot \\
& \Big(\mathbb{I}_{\left[-\frac{\pi}{3},\theta_1-\theta_B\right]}(\theta_4) + \mathbb{I}_{[\theta_1+\theta_B,\theta_3-\theta_B]}(\theta_4) + \mathbb{I}_{[\theta_3+\theta_B,\theta_2-\theta_B]}(\theta_4) \\
& \quad + \mathbb{I}_{\left[\theta_2+\theta_B,\frac{\pi}{3}\right]}(\theta_4) \Big) \;\cdot\; d\theta_1\; d\theta_2\; d\theta_3\; d\theta_4 \\
& + \int_{-\frac{\pi}{3}}^{\frac{\pi}{3}} \int_{\theta_1+\theta_B}^{\frac{\pi}{3}} \int_{\theta_2+\theta_B}^{\frac{\pi}{3}} \int_{-\frac{\pi}{3}}^{\frac{\pi}{3}} T_\Theta(\theta_1)\; T_\Theta(\theta_2)\; T_\Theta(\theta_3)\; T_\Theta(\theta_4) \;\cdot \\
& \Big(\mathbb{I}_{\left[-\frac{\pi}{3},\theta_1-\theta_B\right]}(\theta_4) + \mathbb{I}_{[\theta_1+\theta_B,\theta_2-\theta_B]}(\theta_4) + \mathbb{I}_{[\theta_2+\theta_B,\theta_3-\theta_B]}(\theta_4) \\
& \quad + \mathbb{I}_{\left[\theta_3+\theta_B,\frac{\pi}{3}\right]}(\theta_4) \Big) \;\cdot\; d\theta_1\; d\theta_2\; d\theta_3\; d\theta_4 \qquad \text{(C.34)}
\end{aligned}
$$

$$
\begin{aligned}
I_2 \;=\; & \int_{-\frac{\pi}{3}}^{\frac{\pi}{3}} \int_{\theta_1+\theta_B}^{\frac{\pi}{3}} \int_{-\frac{\pi}{3}}^{\theta_1-\theta_B} T_\Theta(\theta_1)\; T_\Theta(\theta_2)\; T_\Theta(\theta_3)\;\cdot \\
& \qquad\qquad \frac{3}{2\pi}\left[\frac{2\pi}{3}-6\theta_B\right]\; d\theta_1\; d\theta_2\; d\theta_3 \\
& + \int_{-\frac{\pi}{3}}^{\frac{\pi}{3}} \int_{\theta_1+\theta_B}^{\frac{\pi}{3}} \int_{\theta_1+\theta_B}^{\theta_2-\theta_B} T_\Theta(\theta_1)\; T_\Theta(\theta_2)\; T_\Theta(\theta_3)\;\cdot \\
& \qquad\qquad \frac{3}{2\pi}\left[\frac{2\pi}{3}-6\theta_B\right]\; d\theta_1\; d\theta_2\; d\theta_3 \\
& + \int_{-\frac{\pi}{3}}^{\frac{\pi}{3}} \int_{\theta_1+\theta_B}^{\frac{\pi}{3}} \int_{\theta_2+\theta_B}^{\frac{\pi}{3}} T_\Theta(\theta_1)\; T_\Theta(\theta_2)\; T_\Theta(\theta_3)\;\cdot \\
& \qquad\qquad \underbrace{\frac{3}{2\pi}\left[\frac{2\pi}{3}-6\theta_B\right]}_{=1-\frac{9\theta_B}{\pi}}\; d\theta_1\; d\theta_2\; d\theta_3 \qquad \text{(C.35)}
\end{aligned}
$$

$$
\begin{aligned}
= \; & \left(1-\frac{9\theta_B}{\pi}\right) \int_{-\frac{\pi}{3}}^{\frac{\pi}{3}} \int_{\theta_1+\theta_B}^{\frac{\pi}{3}} T_\Theta(\theta_1)\; T_\Theta(\theta_2)\;\cdot \\
& \qquad\qquad \underbrace{\frac{3}{2\pi}\left[\frac{2\pi}{3}-4\theta_B\right]}_{=1-\frac{6\theta_B}{\pi}}\; d\theta_1\; d\theta_2 \qquad \text{(C.36)}
\end{aligned}
$$

$$
= \left(1-\frac{9\theta_B}{\pi}\right)\left(1-\frac{6\theta_B}{\pi}\right) \int_{-\frac{\pi}{3}}^{\frac{\pi}{3}} T_\Theta(\theta_1)\; \underbrace{\frac{3}{2\pi}\left[\theta_B-\frac{\pi}{3}\right]}_{=\left(1-\frac{3\theta_B}{\pi}\right)\;\frac{1}{2}}\; d\theta_1 \qquad \text{(C.37)}
$$

$$
= \left(1-\frac{9\theta_B}{\pi}\right)\left(1-\frac{6\theta_B}{\pi}\right)\left(1-\frac{3\theta_B}{\pi}\right)\frac{1}{2}\,\frac{3}{2\pi}\,\frac{2\pi}{3} \qquad \text{(C.38)}
$$

$$
= \frac{1}{2}\left(1-\frac{9\theta_B}{\pi}\right)\left(1-\frac{6\theta_B}{\pi}\right)\left(1-\frac{3\theta_B}{\pi}\right) \qquad \text{(C.39)}
$$

As a result, we finally obtain that

$$
P[\,!\,4 \text{ beams}] = I_1 + I_2 = \left(1-\frac{9\theta_B}{\pi}\right)\left(1-\frac{6\theta_B}{\pi}\right)\left(1-\frac{3\theta_B}{\pi}\right) \qquad \text{(C.40)}
$$

Considering 4 transmit antennas, $\theta_B = \frac{\pi}{12}$, such as

$$P[!\,4 \text{ beams}] = \left(1 - \frac{3}{4}\right)\ \left(1 - \frac{1}{2}\right)\ \left(1 - \frac{1}{4}\right) = \frac{3}{32} \tag{C.41}$$

C.1.4 Summary

Finally, here are the probabilities of existence of non-overlapping beams we obtained

$$\begin{aligned} P[!\,2 \text{ beams}] &= \left(1 - \tfrac{3\theta_B}{\pi}\right) \\ P[!\,3 \text{ beams}] &= \left(1 - \tfrac{3\theta_B}{\pi}\right)\ \left(1 - \tfrac{6\theta_B}{\pi}\right) \\ P[!\,4 \text{ beams}] &= \left(1 - \tfrac{3\theta_B}{\pi}\right)\ \left(1 - \tfrac{6\theta_B}{\pi}\right)\ \left(1 - \tfrac{9\theta_B}{\pi}\right) \end{aligned} \tag{C.42}$$

Actually, it is easy to generalize these probabilities to

$$P[!\,B \text{ beams}] = \prod_{k=1}^{B-1} \left(1 - k\frac{3\theta_B}{\pi}\right) \tag{C.43}$$

paying attention that a negative value actually corresponds to a null probability.

Considering 4 transmit antennas, $\theta_B = \frac{\pi}{12}$, these probabilities are

$$\begin{aligned} P[!\,2 \text{ beams}] &= \tfrac{3}{4} &&= 0.75 \\ P[!\,3 \text{ beams}] &= \tfrac{3}{8} &&= 0.375 \\ P[!\,4 \text{ beams}] &= \tfrac{3}{32} &&= 0.094 \end{aligned} \tag{C.44}$$

C.2 Probability to find users with non-overlapping beams in a circular cell

As the previous section evaluated the probability of non-overlapping beams, this section analyzes the probability to find among N_{queued} users two, three or four of them towards whom we can steer non-overlapping beams. To simplify computations, we assume here that users have an uniform angular distribution. The following section considers a hexagonal cell distribution with densities of Appendix B.

C.2.1 Probability to find two users

Considering N_{queued} users, there are $C^2_{N_{queued}}$ pairs of users. From a binomial distribution, where success is the existence of two non-overlapping beams, with probability $P[\,!\,2 \text{ beams}] = \frac{3}{4}$, we have that the probability of existence of exactly k pairs of users with non-overlapping beams is

$$P[\,\exists!\, k \text{ pairs}] = C^k_{C^2_{N_{queued}}} \left(\frac{3}{4}\right)^k \left(1-\frac{3}{4}\right)^{C^2_{N_{queued}}-k} \tag{C.45}$$

So the probability that there is not any pair is

$$P[\,\nexists \text{ pair}] = C^0_{C^2_{N_{queued}}} \left(\frac{3}{4}\right)^0 \left(1-\frac{3}{4}\right)^{C^2_{N_{queued}}} \tag{C.46}$$

$$= \left(1-\frac{3}{4}\right)^{C^2_{N_{queued}}} = \left(1-\frac{3}{4}\right)^{\frac{N_{queued}!}{2(N_{queued}-2)!}} \tag{C.47}$$

$$= \left(1-\frac{3}{4}\right)^{\frac{N_{queued}(N_{queued}-1)}{2}} \tag{C.48}$$

As a result, the probability that there exists at least one pair is

$$P[\,!\,1 \text{ pair}] = 1 - P[\,\nexists \text{ pair}] = 1 - \left(1-\frac{3}{4}\right)^{\frac{N_{queued}(N_{queued}-1)}{2}} \tag{C.49}$$

Table C.1 lists the probability to find at least one pair depending on the number N_{queued} of users, assuming uniform angular distribution and $\theta_B = \frac{\pi}{12}$. As one can see, when there are more than 4 users, we are nearly sure to find at least one pair.

N_{queued}	2	3	4	5	$\cdots$
$P[\,!\,1 \text{ pair}]$	0.7500	0.9844	0.9998	1.0000	$\cdots$

Table C.1: Probability to find two users with non-overlapping beams

C.2.2 Probability to find three users

There are $C^3_{N_{queued}}$ triples of users. As for the pairs, considering a binomial distribution, where success is the existence of three non-overlapping beams, with probability $P[\,!\,3 \text{ beams}] = \frac{3}{8}$, we have that the probability of existence of exactly k triples of users with non overlapping beams is

$$P[\,\exists!\, k \text{ triples}] = C^k_{C^3_{N_{queued}}} \left(\frac{3}{8}\right)^k \left(1-\frac{3}{8}\right)^{C^3_{N_{queued}}-k} \tag{C.50}$$

So the probability that there exists at least one triple is

$$P[\,!\,1 \text{ triple}] = 1 - P[\,\nexists \text{ triple}] \tag{C.51}$$

$$= 1 - \left(1 - \frac{3}{8}\right)^{\frac{C^3_{N_{queued}}(C^3_{N_{queued}}-1)(C^3_{N_{queued}}-2)}{6}} \tag{C.52}$$

Table C.2 lists the probability to find at least one triple depending on the number N_{queued} of users, assuming uniform angular distribution and $\theta_B = \frac{\pi}{12}$. As one can see, when there are more than 6 users, we are nearly sure to find at least one triple.

N_{queued}	3	4	5	6	7	$\cdots$
$P[\,!\,1$ triple$]$	0.3750	0.8474	0.9909	0.9999	1.0000	$\cdots$

Table C.2: Probability to find three users with non-overlapping beams

C.2.3 Probability to find four users

There are $C^4_{N_{queued}}$ 4-uples of users. The probability of existence of exactly k 4-uples of users with non overlapping beams is

$$P[\,\exists!\, k \text{ 4-uple}] = C^k_{C^4_{N_{queued}}} \left(\frac{3}{32}\right)^k \left(1 - \frac{3}{32}\right)^{C^4_{N_{queued}}-k} \tag{C.53}$$

So the probability that there exists at least one 4-uple is

$$P[\,!\,1 \text{ 4-uple}] = 1 - P[\,\nexists \text{ 4-uple}] \tag{C.54}$$

$$= 1 - \left(1 - \frac{3}{32}\right)^{\frac{C^4_{N_{queued}}(C^4_{N_{queued}}-1)(C^4_{N_{queued}}-2)(C^4_{N_{queued}}-3)}{24}} \tag{C.55}$$

Table C.3 lists the probability to find at least one 4-uple depending on the number N_{queued} of users, assuming uniform angular distribution and $\theta_B = \frac{\pi}{12}$. As one can see, when there are more than 8 users, we are nearly sure to find at least one 4-uple.

N_{queued}	4	5	6	7	8	9	$\cdots$
$P[\,!\,1$ 4-uple$]$	0.0938	0.3887	0.7716	0.968	0.999	1.000	$\cdots$

Table C.3: Probability to find four users with non-overlapping beams

C.3 Probability to find users with non-overlapping beams in a hexagonal cell

Previous sections of this appendix consider an uniform users angular density, to simplify calculations. But in hexagonal cells, as we have shown in Appendix B, their distribution is not uniform. We therefore have to evaluate the probability of overlapping beams with non-uniform beam density, and then to assess the probability to find some users among the N_{queued} ones that can have non-overlapping beams.

Due to calculation hardness, we only derive closed-form expressions of probabilities for two simultaneous beams. Results are actually quite consistent with those obtained considering uniform angular density, so calculations for three and four simultaneous beams are not regarded as necessary.

C.3.1 Probability that 2 beams are not overlapping

Here is the angular density in a hexagonal cell sector for a uniform user repartition we obtained in Appendix B:

$$T_\Theta(\theta) = \begin{cases} \frac{3\sqrt{3}}{\left(3\cos\theta+\sqrt{3}\sin\theta\right)^2} & \text{if} \quad \theta \geq 0 \\ \frac{3\sqrt{3}}{\left(3\cos\theta-\sqrt{3}\sin\theta\right)^2} & \text{if} \quad \theta < 0 \end{cases} \tag{C.56}$$

The probability that two random beams are not overlapping each other is obtained like in Eq. (C.4), from

$$P[\,!\,2 \text{ beams}] = \int_{-\frac{\pi}{3}}^{\frac{\pi}{3}} \int_{-\frac{\pi}{3}}^{\frac{\pi}{3}} T_\Theta(\theta_1)\; T_\Theta(\theta_2)$$
$$\left(\mathbb{I}_{\left[-\frac{\pi}{3}\,\theta_1-\theta_B\right]}(\theta_2) + \mathbb{I}_{\left[\theta_1+\theta_B\,\frac{\pi}{3}\right]}(\theta_2)\right) d\theta_2 d\theta_1 \tag{C.57}$$
$$= \int_{-\frac{\pi}{3}}^{\frac{\pi}{3}} T_\Theta(\theta_1)$$
$$\underbrace{\int_{-\frac{\pi}{3}}^{\frac{\pi}{3}} T_\Theta(\theta_2)\;\left(\mathbb{I}_{\left[-\frac{\pi}{3}\,\theta_1-\theta_B\right]}(\theta_2) + \mathbb{I}_{\left[\theta_1+\theta_B\,\frac{\pi}{3}\right]}(\theta_2)\right) d\theta_2}_{=1-\int_{\theta_1-\theta_B}^{\theta_1+\theta_B} T_\Theta(\theta_2)d\theta_2 \text{ since } T_\Theta \text{ is a density function over } \left[-\frac{\pi}{3},\frac{\pi}{3}\right]} \quad d\theta_1 \tag{C.58}$$
$$= \underbrace{\int_{-\frac{\pi}{3}}^{\frac{\pi}{3}} T_\Theta(\theta_1)d\theta_1}_{=1 \text{ (density function)}} - \int_{-\frac{\pi}{3}}^{\frac{\pi}{3}} T_\Theta(\theta_1) \int_{\theta_1-\theta_B}^{\theta_1+\theta_B} T_\Theta(\theta_2)\; d\theta_2\; d\theta_1 \tag{C.59}$$
$$= 1 - \int_{-\frac{\pi}{3}}^{\frac{\pi}{3}} T_\Theta(\theta_1) \int_{\theta_1-\theta_B}^{\theta_1+\theta_B} T_\Theta(\theta_2)\; d\theta_2\; d\theta_1 \tag{C.60}$$

If $\theta_1 \in \left[-\frac{\pi}{3},\, \theta_B\right]$, from [85, (2.557.5) p.170] we have

$$\int_{\theta_1-\theta_B}^{\theta_1+\theta_B} T_\Theta(\theta_2)\; d\theta_2 \;=\; \int_{\theta_1-\theta_B}^{\theta_1+\theta_B} \frac{\sqrt{3}\; d\theta_2}{\left(\sqrt{3}\cos\theta_2 - \sin\theta_2\right)^2} \tag{C.61}$$

$$= \frac{\sqrt{3}}{4}\left[\frac{\sqrt{3}\sin\theta_2 + \cos\theta_2}{\sqrt{3}\cos\theta_2 - \sin\theta_2}\right]_{\theta_1-\theta_B}^{\theta_1+\theta_B} \tag{C.62}$$

Let us write

$$\left\{\begin{array}{lcl} a &=& \theta_1 - \theta_B \\ b &=& \theta_1 + \theta_B \end{array}\right. \Rightarrow \left\{\begin{array}{lcll} a+b &=& 2\theta_1 & \\ b-a &=& 2\theta_B &= \frac{\pi}{6} \end{array}\right. \tag{C.63}$$

So $\displaystyle\int_{\theta_1-\theta_B}^{\theta_1+\theta_B} T_\Theta(\theta_2)\; d\theta_2$

$$= \frac{\sqrt{3}}{4}\left(\frac{\sqrt{3}\sin b + \cos b}{\sqrt{3}\cos b - \sin b} - \frac{\sqrt{3}\sin a + \cos a}{\sqrt{3}\cos a - \sin a}\right) \tag{C.64}$$

$$= \frac{\sqrt{3}}{4}\left(\frac{4\cos a\sin b - 4\cos b\sin a}{\left(\sqrt{3}\cos b - \sin b\right)\cdot\left(\sqrt{3}\cos a - \sin a\right)}\right) \tag{C.65}$$

$$= \sqrt{3}\;\frac{\sin(b-a)}{-\sqrt{3}\sin(a+b) + \cos(a+b) + 2\cos(b-a)} \tag{C.66}$$

$$= \sqrt{3}\;\frac{\sin\frac{\pi}{6}}{-\sqrt{3}\sin(2\theta_1) + \cos(2\theta_1) + 2\cos\frac{\pi}{6}} \tag{C.67}$$

$$= \frac{\frac{\sqrt{3}}{2}}{\sqrt{3} - \sqrt{3}\sin(2\theta_1) + \cos(2\theta_1)} \tag{C.68}$$

If $\theta_1 \in [-\theta_B,\, \theta_B]$,

$$\int_{\theta_1-\theta_B}^{\theta_1+\theta_B} T_\Theta(\theta_2)\; d\theta_2$$

$$= \int_{\theta_1-\theta_B}^{0} \frac{\sqrt{3}\; d\theta_2}{\left(\sqrt{3}\cos\theta_2 - \sin\theta_2\right)^2} + \int_{0}^{\theta_1+\theta_B} \frac{\sqrt{3}\; d\theta_2}{\left(\sqrt{3}\cos\theta_2 + \sin\theta_2\right)^2} \tag{C.69}$$

$$= \frac{\sqrt{3}}{4}\left[\frac{\sqrt{3}\sin\theta_2 - \cos\theta_2}{\sqrt{3}\cos\theta_2 + \sin\theta_2}\right]_{\theta_1-\theta_B}^{0} + \frac{\sqrt{3}}{4}\left[\frac{\sqrt{3}\sin\theta_2 - \cos\theta_2}{\sqrt{3}\cos\theta_2 + \sin\theta_2}\right]_{0}^{\theta_1+\theta_B} \tag{C.70}$$

So

$$\int_{\theta_1-\theta_B}^{\theta_1+\theta_B} T_\Theta(\theta_2)\, d\theta_2$$

$$= \frac{\sqrt{3}}{4}\left(\frac{1}{\sqrt{3}} - \frac{\sqrt{3}\sin a + \cos a}{\sqrt{3}\cos a - \sin a} + \frac{\sqrt{3}\sin b - \cos b}{\sqrt{3}\cos b + \sin b} + \frac{-1}{\sqrt{3}}\right) \quad \text{(C.71)}$$

$$= \frac{1}{2} + \frac{\sqrt{3}}{4}\left(\frac{2\sin(b-a) - 2\sqrt{3}\cos(b-a)}{2\cos(a+b) + cos(b-a) + \sqrt{3}\sin(b-a)}\right) \quad \text{(C.72)}$$

$$= \frac{1}{2} - \frac{\frac{\sqrt{3}}{2}}{\sqrt{3} + 2\cos 2\theta_1} \quad \text{(C.73)}$$

And finally if $\theta_1 \in \left[\theta_B,\, \frac{\pi}{3}\right]$,

$$\int_{\theta_1-\theta_B}^{\theta_1+\theta_B} T_\Theta(\theta_2)\, d\theta_2 = \sqrt{3}\int_{\theta_1-\theta_B}^{\theta_1+\theta_B} \frac{d\theta_2}{\left(\sqrt{3}\cos\theta_2 + \sin\theta_2\right)^2} \quad \text{(C.74)}$$

$$= \frac{\sqrt{3}}{4}\left[\frac{\sqrt{3}\sin\theta_2 - \cos\theta_2}{\sqrt{3}\cos\theta_2 + \sin\theta_2}\right]_{\theta_1-\theta_B}^{\theta_1+\theta_B} \quad \text{(C.75)}$$

From similar calculations as if $\theta_1 \in \left[-\frac{\pi}{3},\, \theta_B\right]$, we obtain

$$\int_{\theta_1-\theta_B}^{\theta_1+\theta_B} T_\Theta(\theta_2)\, d\theta_2 = \frac{\frac{\sqrt{3}}{2}}{\sqrt{3} + \sqrt{3}\sin(2\theta_1) + \cos(2\theta_1)} \quad \text{(C.76)}$$

As a result, we finally have that

$P[!\,2 \text{ beams}] = 1$

$$-\int_{-\frac{\pi}{3}}^{-\theta_B} \frac{3\sqrt{3}}{\left(3\cos\theta_1 - \sqrt{3}\sin\theta_1\right)^2} \frac{\frac{\sqrt{3}}{2}}{\sqrt{3} - \sqrt{3}\sin(2\theta_1) + \cos(2\theta_1)}\, d\theta_1$$

$$-\int_{-\theta_B}^{0} \frac{3\sqrt{3}}{\left(3\cos\theta_1 - \sqrt{3}\sin\theta_1\right)^2} \left(\frac{1}{2} - \frac{\frac{\sqrt{3}}{2}}{\sqrt{3} + 2\cos 2\theta_1}\right) d\theta_1$$

$$-\int_{0}^{\theta_B} \frac{3\sqrt{3}}{\left(3\cos\theta_1 + \sqrt{3}\sin\theta_1\right)^2} \left(\frac{1}{2} - \frac{\frac{\sqrt{3}}{2}}{\sqrt{3} + 2\cos 2\theta_1}\right) d\theta_1$$

$$-\int_{\theta_B}^{\frac{\pi}{3}} \frac{3\sqrt{3}}{\left(3\cos\theta_1 + \sqrt{3}\sin\theta_1\right)^2} \frac{\frac{\sqrt{3}}{2}}{\sqrt{3} + \sqrt{3}\sin(2\theta_1) + \cos(2\theta_1)}\, d\theta_1 \quad \text{(C.77)}$$

From numerical approximations performed with MATLAB, we obtained

$$P[\,!\,2 \text{ beams}] = 1 - 0.1064 - 0.0352 - 0.0352 - 0.1064 = 0.7168 \quad \text{(C.78)}$$

which is quite close from 0.75 obtained with uniform angular density.

C.3.2 Probability to find 2 users with non-overlapping beams

Considering the $C^2_{N_{queued}}$ pairs of users, the probability of existence of exactly k pairs of users with non-overlapping beams is

$$P[\,\exists!\, k \text{ pairs}] \;=\; C^k_{C^2_{N_{queued}}} \; (0.7168)^k \; (1 - 0.7168)^{C^2_{N_{queued}} - k} \quad \text{(C.79)}$$

So the probability that there exists at least one pair is

$$P[\,!\,1 \text{ pair}] \;=\; 1 - P[\,\nexists \text{ pair}] = 1 - \Big(1 - 0.7168\Big)^{\frac{N_{queued}(N_{queued}-1)}{2}} \quad \text{(C.80)}$$

Table C.4 lists the probability to find at least one pair depending on the number N_{queued} of users, assuming uniform surface distribution and $\theta_B = \frac{\pi}{12}$. As one can see, when there are more than 4 users, we are nearly sure to find at least one pair. Moreover, probabilities obtained here are pretty close from those obtained considering uniform angular density.

N_{queued}	2	3	4	5	$\cdots$
$P[\,!\,1 \text{ pair}]$	0.7168	0.9773	0.9995	1.0000	$\cdots$

Table C.4: Probability to find two users with non-overlapping beams

Bibliography

[1] James F. Kurose and Keith W. Ross, *Computer Networking - A Top-Down Approach Featuring the Internet (5th Edition)*, Addison Wesley, 2009.

[2] E. C. Strinati, G. Corbellini and D. Ktenas, "HYGIENE Scheduling for OFDMA Wireless Cellular Networks", *Proc. of IEEE 69th Vehicular Technology Conference VTC 2009 Spring*, vol. 13, April 2009.

[3] Mobile WiMAX Part I, "A Technical Overview and Performance Evaluation", Technical report, WiMAX Forum, April 2006.

[4] M. Andrews, K. Kumaran, K. Ramanan, A. Stolyar and P. Whiting, "Providing quality of service over a shared wireless link", *IEEE Communications Magazine*, vol. 39, pp. 150–154, 2001.

[5] Harri Holma and Antti Toskala, *HSDPA/HSUPA for UMTS*, John Wiley & Sons, Ltd., 2006.

[6] Mario Kountounis, Ashish Pandharipande, Hojin Kim and David Gesbert, "QoS-based User Scheduling for Multiuser MIMO Systems", *Proceedings of IEEE Vehicular Techonolgy Conference, VTC 1999 Spring*, 1999.

[7] H. Fattah and C. Leung, "An overview of scheduling algorithms in wireless multimedia networks", *IEEE Wireless Communications*, vol. 9, n. 5, pp. 76–83, 2002.

[8] W. Ajib and D. Haccoun, "An overview of scheduling algorithms in mimo-based fourth-generation wireless systems", *IEEE Network*, vol. 19, n. 5, pp. 43–48, 2005.

[9] S. Shakkottai and A. Stoylar, "Scheduling algorithms for a mixture of real-time and non-real-time data in hdr", *17th Int. Teletraffic Congress*, 2001.

[10] Raymond Kwan, Peter H.G. Chong, Eeva Poutiainen and Mika Rinne, "The Effect of Code-Multiplexing on the High Speed Downlink Packet Access (HSDPA) in a WCDMA Network", in *Proceedings of IEEE Wireless Communications and Networking Conference WCNC'03, New Orleans (LA), United States of America*, pp. 1728–1732, March 2003.

[11] Harri Holma and Antti Toskala, *WCDMA For UMTS HSDPA Evolutions and LTE*, John Wiley & Sons, Ltd., 2007.

[12] Lars Torsten Berger, *Performance of Multi-Antenna Enhanced HSDPA*, PhD thesis, Aalborg University, Denmark, May 2005.

[13] 3rd Generation Partnership Project, "Physical Layer Aspects for Evolved Universal Terrestrial Radio Access", Technical Report 3GPP TR 25.814 v7.1.0, September 2006.

[14] C.E. Shannon, "A Mathematical Theory of Communication", *Bell System Technology Journal*, vol. 27, 1948.

[15] N. Mehta, Andreas F. Molisch and L. J. Greenstein, "Orthogonality Factor in WCDMA Downlinks in Urban Macrocellular Environments", *Proceedings of IEEE Global Telecommunications Conference Globecom*, 2005.

[16] Hugues Van Peteghem, *Builing a Testbed Emulating Cellular Networks*, PhD thesis, The University of Namur, Computer Science Faculty, Belgium, March 2008.

[17] Taesang Yoo, *Sum-Capacity, Scheduling, and Multi-user Diversity in MIMO Broadcast Systems*, PhD thesis, Stanford University, September 2007.

[18] R Karp, "Reducibility among combinatorial problems", *Complexity of Computer Computations, R.E. Miller and J.W. Thatcher (eds.), Plenum Press, New York*, pp. 85–104, 1972.

[19] Ding-Zhu Du and P.M. Pardalos, *Handbook of Combinatorial Optimization Supplement Volume A*, Springer, 1999.

[20] S. Vangipuram and S. Bhashyam, "Multiuser Scheduling and Power Sharing for CDMA Packet Data Systems", *Proceedings of IEEE NCC Conference*, 2007.

[21] Harri Holma and Antti Toskala, *WCDMA for UMTS - HSDPA Evolutions and LTE*, John Wiley & Sons, Ltd., 2007.

[22] Jari Salo, Giovanni Del Galdo, Jussi Salmi, Pekka Kyösti, Marko Milojevic, Daniela Laselva and Christian Schneider, "MATLAB implementation of the 3GPP Spatial Channel Model (3GPP TR 25.996)", Technical report, European IST-2003-507591 Project WINNER, http://www.ist-winner.org, January 2005.

[23] R. Jain, D.M. Chiu and W. Hawe, "A Quantitative Mesure of Fairness and Discrimination for Resource Allocation in Shared Systems", *DEC Research Report TR-301.1984*, 1984.

[24] Raymond Knopp and Pierre Humblet, "Multiple-accessing over frequency-selective fading channels", in *PIMRC 1995, 6th IEEE International Symposium on Personal, Indoor and Mobile Radio Communications, September 27-29, 1995, Toronto, Canada*, 10 1995.

[25] J. Vanderpypen and L. Schumacher, "Multistream Proportional Fair Packet Scheduling Optimization in HSDPA", *Proceedings of the 15th Annual Symposium of the IEEE/CVT Benelux Chapter SCVT 2008*, 2008.

[26] H. Schulze and C. Lüders, *Theory and Application of OFDMA and CDMA*, John Wiley & Sons, Ltd., 2005.

[27] V. Tarokh, H. Jafarkhani and A.R. Calderbank, "Space-time block codes from orthogonal designs", *Information Theory, IEEE Transactions on*, vol. 45, n. 5, pp. 1456 –1467, jul 1999.

[28] I. Emre Telatar, "Capacity of Multi-Antenna Gaussian Channels", *European Transactions on Telecommunications*, vol. 10, n. 6, November 1999.

[29] P. W. Wolniansky, Gerard J. Foschini, G. D. Golden and Reinaldo A. Valenzuela, "V-BLAST: An Architecture for Realizing Very High Data Rates over the Rich-Scattering Wireless Channel", in *Proceedings of International Symposium on Advanced Radio Technologies, Boulder, CO, USA*, September 1998.

[30] P. Stoica and R. Moses, *Spectral Analysis of Signal*, Pearson Prentice Hall, 2005.

[31] Fang Shu, Li Lihua, Tao Xiaofeng and Zhang Ping, "A Spatial Multiplexing MIMO Scheme with Beamforming for Downlink Transmission", in *Vehicular Technology Conference, 2007. VTC-2007 Fall. 2007 IEEE 66th*, pp. 700 –704, 30 2007-oct. 3 2007.

[32] Zhongding Lei, F.P.S. Chin and Ying-Chang Liang, "Orthogonal switched beams for downlink diversity transmission", *Antennas and*

Propagation, IEEE Transactions on, vol. 53, n. 7, pp. 2169 – 2177, july 2005.

[33] Fan Zhu, Kyung Sik Ryu and Myoung Seob Lim, "Combined beamforming with space-time block coding using double antenna array group", in *Spread Spectrum Techniques and Applications, 2004 IEEE Eighth International Symposium on*, pp. 755 – 758, aug.-2 sept. 2004.

[34] Lin lin Wang, Shu xun Wang, Xiao ying Sun and Feng ye Hu, "Combined beamforming and space-time block coding for wireless communications", in *Personal, Indoor and Mobile Radio Communications, 2003. PIMRC 2003. 14th IEEE Proceedings on*, volume 1, pp. 607 – 611 Vol.1, sept. 2003.

[35] Lizhong Zheng and David Tse, "Diversity and Multiplexing: A Fundamental Trade-Off in Multiple Antenna Channels", *IEEE Transactions on Information Theory*, vol. 49, n. 5, 2003.

[36] U. Spagnolini, "A Simplified Model to Evaluate the Probability of Error in DS-CDMA Systems", *IEEE Transactions on Wireless Communication*, vol. 3, n. 2, 2004.

[37] P. Viswanath, D.N.C. Tse and R. Laroia, "Opportunistic beamforming using dumb antennas", *Information Theory, IEEE Transactions on*, vol. 48, n. 6, pp. 1277 –1294, jun 2002.

[38] Patrick Svedman, Sarah Kate Wilson, Leonard J. Cimini and Björn Ottersten, "Opportunistic Beamforming and Scheduling for OFDMA Systems", *IEEE Transactions on Communications*, vol. 55, n. 5, pp. 941–952, May 2007.

[39] Man-On Pun, Kyeong Jim Kim, Roland A. Iltis and H. Vincent Poor, "Reduced-feedback opportunistic scheduling and beamforming with GMD for MIMO-OFDMA", *Proc. of 42nd Asilomar Conference on Signals, Systems and Computers*, 2008.

[40] Goran Dimic and Nicholas D. Sidiropoulos, "On downlink beamforming with greedy user selection: performance analysis and a simple new algorithm", *IEEE Transactions on Signal Processing*, vol. 53, n. 10, pp. 3857–3868, octobre 2005.

[41] Tsaesang Yoo, Nihar Jindal and Andrea Goldsmith, "Multi-Antenna Downlink Channels with Limited Feedback and User Selection", *IEEE Journal of Selected Areas in Communications*, vol. 25, n. 7, pp. 1478–1491, September 2007.

[42] Ermanna Conte, Stefano Tomasin and Nevio Benvenuto, "A Simplified Greedy Algorithm for Joint Scheduling and Beamforming in Multiuser MIMO OFDM", *IEEE Communications Letters*, vol. 14, n. 5, pp. 381–383, May 2010.

[43] D.J. Love, Jr. Heath, R.W. and T. Strohmer, "Grassmannian beamforming for multiple-input multiple-output wireless systems", *IEEE Transactions on Information Theory*, vol. 49, n. 10, pp. 2735–2747, October 2003.

[44] WINNER WP2 Radio Interface, "D2.7 v1.1 Assessment of Advanced Beamforming and MIMO Technologies", Technical report, IST-2003-507581 WINNER, February 2005.

[45] J. Vanderpypen and L. Schumacher, "Multistream Proportional Fair Scheduling Applied on Beamforming Technologies", COST 2100 TD(08) 663, FUNDP, Belgium, October 2008.

[46] D. Astely, E. Dahlman, A. Furuskar, Y. Jading, M. Lindstrom and S. Parkvall, "LTE: the evolution of mobile broadband", *Communications Magazine, IEEE*, vol. 47, n. 4, pp. 44–51, April 2009.

[47] Yanqun Le, Yi Wu and Dongmei Zhang, "An Improved Scheduling Algorithm for rtPS Services in IEEE 802.16", in *Proceedings of Vehicular Technology Conference VTC Spring 2009, Barcelona (Spain)*, 2009.

[48] A. Larmo, M. Lindstrom, M. Meyer, G. Pelletier, J. Torsner and H. Wiemann, "The LTE Link-Layer Design", *Communications Magazine, IEEE*, vol. 47, n. 4, pp. 52–59, April 2009.

[49] Lihua Wan, Wenchao Ma and Zihua Guo, "A Cross-layer Packet Scheduling and Subchannel Allocation Scheme in 802.16e OFDMA System", in *Proceedings of IEEE Wireless Communications and Networking Conference WCNC 2007, Hong Kong (China)*, 2007.

[50] M. Andrews and L. Zhang, "Scheduling Algorithms for Multicarrier Wireless Data Systems", *Networking, IEEE/ACM Transactions on*, vol. 19, n. 2, pp. 447 –455, april 2011.

[51] W. Rhee and J. Cioffi, "Increase in Capacity of Multiuser OFDM System Using Dynamic Subchannel Allocation", *Proceedings of IEEE Vehicular Techonolgy Conference, VTC 2000 Spring, Tokyo, Japan*, vol. 2, pp. 1085–1089, May 2000.

[52] A. Gotsis, D Komnakos and P Constantinou, "Dynamic Subchannel and Slot Allocation for OFDMA Networks Supporting Mixed

Traffic: Upper Bound and a Heuristic Algorithm", *IEEE Communication Letters*, vol. 13, n. 8, pp. 576–578, August 2009.

[53] G. Yu, Z. Zhang and P. Qiu, "Adaptative Subcarrier and Bit Allocation in OFDMA Systems Supporting Heterogeneous Services", *J. Wireless Personal Commun. (Springer, Netherlands)*, vol. 43, n. 4, pp. 1057–1070, December 2007.

[54] B Sadiq, R. Madan and A. Sampath, "Downlink Scheduling for Multi-class Traffic in LTE", *EURASIP Journal on Wireless Communications and Networking*, 2009.

[55] Amos Israeli, Dror Rawitz and Oran Sharon, "On the complexity of sequential rectangle placement in IEEE 802.16/WiMAX systems", *Inf. Comput.*, vol. 206, pp. 1334–1345, November 2008.

[56] R. Cohen and L. Katzir, "Computational Analysis and Efficient Algorithms for Micro and Macro OFDMA Scheduling", in *The 27th Conf. on Computer Comm. INFOCOM 2008.*, pp. 511–519, 2008.

[57] 3GPP2 TSG-RAN148, "LTE physical layer framework for performance verification", Technical report, February 2007.

[58] Xiaoxin Wu, Juejia Zhou, Guangjie Li and May Wu, "Low Overhead CQI Feedback in Multi-Carrier Systems", *Proceedings of IEEE Global Telecommunications Conference Globecom 2007*, November 2007.

[59] H. Zhu and J. Wang, "Chunk-Based Resource Allocation in OFDMA Systems Part I: Chunk Allocation", *IEEE Transactions on Communications*, vol. 57, n. 9, pp. 2734–2744, 2009.

[60] B. Varadarajan, Runhua Chen, E.N. Onggosanusi, Il Han Kim and A.G.; Dabak, "Efficient Channel Quality Feedback Schemes For OFDMA Systems With Different Schedulers", *Proceedings of IEEE 69th Vehicular Technology Conference VTC 2009 Spring*, April 2009.

[61] Masood Maqbool, Marceau Coupechoux and Philippe Godlewski, "Subcarrier Permutation Types in IEEE 802.16e", Technical report, ENST (Télécom Paris), April 2008.

[62] M. A. Awal and Lila Boukhatem, "Opportunistic Periodic Feedback Mechanisms for OFDMA Systems under Feedback Budget Constraint", *Proceedings of IEEE Vehicular Technology Conference VTC 2011 Spring*, 2011.

[63] Chung-Ju Chang, Chih-Ming Yen, Fang-Ching Ren and Jian-Ann Lai, "Dynamic Priority-based Resource Allocation for Uplinks in IEEE 802.16 Wireless Communication Systems", in *Proceedings of Vehicular Technology Conference VTC Fall 2008, Calgary (Canada)*, September 2008.

[64] Antonio Iera, Antonella Molinaro and Sara Pizzi, "Channel-Aware Scheduling for QoS and Fairness Provisioning in IEEE 802.16/WiMAX Broadband Wireless Access Systems", *IEEE Network*, vol. 21, n. 5, pp. 34–41, September 2007.

[65] 3rd Generation Partnership Project; Technical Specification Group Services and System Aspects, "Policing and charging control architecture (Release 8)", Technical Report TS 23.203 v8.6.0, June 2009.

[66] S. Manabe and H. Asai, "A neuro-based optimization algorithm for tiling problems with rotation", *Neural Processing Letters*, vol. 13, n. 3, pp. 267–275, 2001.

[67] Ben Shneiderman, "Tree visualization with tree-maps: 2-d space-filling approach", *ACM Transactions on Graphics*, vol. 11, n. 1, pp. 92–99, January 1992.

[68] Mark Bruls, Kees Huizing and Jarke J. van Wijk, "Squarified Treemaps", in *Proceedings of the Joint Eurographics and IEEE TCVG Symposium on Visualization VisSym'99, Vienna (Austria)*, pp. 33–42, May 1999.

[69] T. Celcer, T. Javornik and G. Kandus, "Fairness Oriented Scheduling Algorithm with QoS Support for Broadband MIMO Systems with Heterogeneous Traffic", COST 2100 TD(09)927, Jozef Stefan Institute, Slovenia, September 2009.

[70] "IEEE Standard 802.16e-2005", Part 16: Air interface for fixed broadband wireless access systems, physical and medium access control layers for combined fixed and mobile operation in licensed bands, 2005.

[71] A. Erta, C. Cicconetti and L. Lenzini, "A downlink data region allocation algorithm for IEEE 802.16e OFDMA", in *6th International Conf. on Information, Communications Signal Processing, 2007*, pp. 1–5, 2007.

[72] H.G. Myung, J Lim and D.J. Goodman, "Single carrier FDMA for uplink wireless transmission", *IEEE Vehicular Technology Magazine*, vol. 1, n. 3, pp. 30 – 38, 2006.

[73] 3rd Generation Partnership Project, "Feasibility Study for OFDM for UTRAN enhancement (Rel. 6)", Technical Report 3GPP TR 25.982 v2.0.0, June 2006.

[74] Thomas Klingenbrunn and Preben E. Mogensen, "Modelling Frequency Correlation of Fast Fading in GSM System and Link Level Simulations", in *Proceedings of IEEE 50th Vehicular Technology Conference VTC 1999 Fall, Amsterdam, The Netherlands*, volume 4, pp. 2398–2402, September 1999.

[75] Joel Vanderpypen, Laurent Schumacher, Javier Daza-Conejero and Gabriel Schwanen, "Tiling Subchannels/PRBs in Mobile WiMAX/LTE", COST 2100 TD(09) 967, FUNDP, Belgium, 2009.

[76] Tarcision F. Maciel and Anja Klein, "On the Performance, Complexity, and Fairness of Suboptimal Resource Allocation for Multiuser MIMO–OFDMA Systems", *IEEE Transactions on Vehicular Technology*, vol. 59, n. 1, pp. 406–419, January 2010.

[77] "3GPP TS36.213", Evolved Universal Terrestrial Radio Access (E-UTRA); physical layer procedures, version 8.5.0, 2005.

[78] S. Martello and P. Toth, *Knapsack problems: Algorithms and computer implementations*, John Wiley & Sons, 1990.

[79] T. Ohseki, M. Morita and T. Inoue, "Burst Construction and Packet Mapping Scheme for OFDMA Downlinks in IEEE 802.16 Systems", in *GLOBECOM '07*, pp. 4307 –4311, 2007.

[80] Y. Ben-Shimol, I. Kitroser and Y. Dinitz, "Two-dimensional mapping for wireless OFDMA systems", *Broadcasting, IEEE Transactions on*, vol. 52, n. 3, pp. 388–396, 2006.

[81] Chakchai So-In, R. Jain and A.-K. Al Tamimi, "OCSA: An algorithm for burst mapping in IEEE 802.16e mobile WiMAX networks", in *15th Asia-Pacific Conference on Communications APCC.*, pp. 52–58, 2009.

[82] C. So-In, R. Jain and A.-K. Al Tamimi, "eOCSA: An algorithm for burst mapping with strict QoS requirements in IEEE 802.16e Mobile WiMAX networks", in *Wireless Days (WD), 2nd IFIP*, pp. 1–5, 2009.

[83] Chakchai So-In, Raj Jain and Abdel-Karim Tamimi, "Capacity evaluation for IEEE 802.16e mobile WiMAX", *J. Comp. Sys., Netw., and Comm.*, vol. 2010, pp. 2:1–2:1, January 2010.

[84] Joel Vanderpypen and Laurent Schumacher, "Treemap-based Burst Mapping Algorithm for Downlink Mobile WiMAX Systems", Submitted for presentation at IEEE Vehicular Technology Conference 2011 Fall.

[85] I.S. Gradshteyn and I.M. Ryzhik, *Table of Integrals, Series, and Products - Sixth Edition*, Academic Press, 2000.